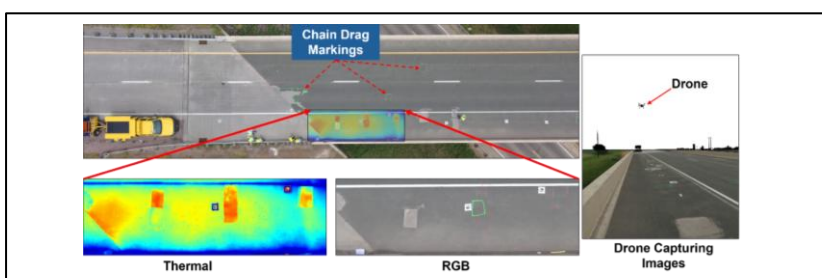




COMPUTER VISION TOOLS FOR BRIDGE INSPECTIONS AND REPORTING – CONCRETE DECKS AND STEEL GIRDERS



Prepared by:

Ammad Khan, Ph.D.^(a)

Abir Mohammad Hadi, Ph.D.^(b)

Kwanghee Won, Ph.D.^(b)

Mostafa Tazarv, Ph.D., P.E.^(a)

Department of Civil and Environmental Engineering^(a)

University of Nevada Reno

Reno, NV 89557

Department of Electrical Engineering and Computer Science^(b)

South Dakota State University

Brookings, SD 57007

Report # FHWA-AK-4000(219)

January 2026

Prepared for:

Alaska Department of Transportation & Public Facilities Statewide Research Office

3132 Channel Drive

Juneau, AK 99801-7898

Publication Number

REPORT DOCUMENTATION PAGE			Form approved OMB No.	
Public reporting for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestion for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-1833), Washington, DC 20503				
1. AGENCY USE ONLY (LEAVE BLANK) FHWA-AK-4000(219)		2. REPORT DATE December 2025		3. REPORT TYPE AND DATES COVERED Final Report, August 2022 through December 2025
4. TITLE AND SUBTITLE Computer Vision Tools for Bridge Inspections and Reporting – Concrete Decks and Steel Girders			5. FUNDING NUMBERS HFHWY00301, 4000(219), 69A3552348308	
6. AUTHOR(S) Ammad Khan ^(a) , Abir Mohammad Hadi ^(b) , Kwanghee Won, Ph.D. ^(b) , Mostafa Tazarv, Ph.D., P.E. ^(a)			8. PERFORMING ORGANIZATION REPORT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Department of Civil and Environmental Engineering ^(a) University of Nevada Reno Reno, NV 89557 Department of Electrical Engineering and Computer Science ^(b) South Dakota State University Brookings, SD 57007				
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) State of Alaska, Alaska Dept. of Transportation and Public Facilities Research and Technology Transfer 3132 Channel Drive Juneau, AK 99801-7898			10. SPONSORING/MONITORING AGENCY REPORT NUMBER FHWY-AK-4000(219)	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT No restrictions			12b. DISTRIBUTION CODE N/A	
13. ABSTRACT (Maximum 200 words) <i>The Alaska Department of Transportation & Public Facilities is responsible for condition assessment of approximately 1,000 bridges. Routine inspections are necessary to monitor condition states of these bridges. In current practice, bridges are inspected mostly manually, and their defects are also quantified manually. The main objective of the present project was to develop practical artificial intelligence (AI) tools that help inspectors with defect detections and quantifications, and to facilitate the inspection reporting following standard requirements. To achieve the project goals, a comprehensive review of bridge deck inspection for delamination with a focus on computer vision, a field of AI, was performed first. Then, more than 13,000 RGB and 2,000 thermal images were collected by inspecting more than 50 bridges. Furthermore, the deck of 18 bridges with delamination were laser scanned using Light Detection and Ranging (lidar) sensors of smart devices. Subsequently, the study proposed three methods to automatically generate inspection reports for concrete bridge decks. In the first method, a three-dimensional (3D) scan of bridge decks is performed using lidar sensors of iPhone/iPad and a two-dimensional (2D) deck map is extracted from the point cloud data. In the second method, a drone equipped with a Red-Green-Blue (RGB) camera is used to collect overlapping images of bridge decks then a 2D deck map is generated. In the third method, a drone equipped with both RGB and thermal cameras is used to collect deck data then 2D RGB-thermal maps of bridge decks are produced. Using large image datasets compiled in the present study, deep neural networks were trained to detect deck delamination by (1) finding inspectors' marking in RGB maps obtained from smartphones or drones, and (2) detecting hot regions in thermal maps. The accuracy of the neural networks to detect delaminated areas of concrete decks in lidar/RGB and thermal images was 96% and 96%, respectively. A human-in-the-loop method was also developed to quantify defects of steel girders using RGB images. Furthermore, guidelines to collect data using different devices were recommended. Finally, a user-friendly AI powered software, Bridge Computer Vision Inspection Tools (Br.CVIT), was developed to facilitate bridge inspections and reporting for concrete decks and steel girders.</i>				
14. KEYWORDS: Automatic Bridge Inspection; Computer Vision; Concrete Deck; Delamination; Nondestructive Evaluation; Drones; Thermal Images; Lidar; Deep Learning; Steel Girders.			15. NUMBER OF PAGES 186	
			16. PRICE CODE N/A	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT N/A	

Notice

This document is disseminated under the sponsorship of the U.S. Department of Transportation in the interest of information exchange. The U.S. Government assumes no liability for the use of the information contained in this document. The U.S. Government does not endorse products or manufacturers. Trademarks or manufacturers' names appear in this report only because they are considered essential to the objective of the document.

Quality Assurance Statement

The Federal Highway Administration (FHWA) provides high-quality information to serve Government, industry, and the public in a manner that promotes public understanding. Standards and policies are used to ensure and maximize the quality, objectivity, utility, and integrity of its information. FHWA periodically reviews quality issues and adjusts its programs and processes to ensure continuous quality improvement.

Author's Disclaimer

Opinions and conclusions expressed or implied in the report are those of the author. They are not necessarily those of the Alaska DOT&PF or funding agencies.

METRIC (SI*) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
<u>LENGTH</u>				
in	inches	25.4	millimeters	mm
ft	feet	0.3048	meters	m
yd	yards	0.914	meters	m
mile	Miles (statute)	1.61	kilometers	km
<u>AREA</u>				
in ²	square inches	645.2	millimeters squared	mm ²
ft ²	square feet	0.0929	meters squared	m ²
yd ²	square yards	0.836	meters squared	m ²
mi ²	square miles	2.59	kilometers squared	km ²
ac	acres	0.4046	hectares	ha
<u>MASS (WEIGHT)</u>				
oz	Ounces (avdp)	28.35	grams	g
lb	Pounds (avdp)	0.454	Kilograms	kg
T	Short tons (2000 lb)	0.907	megagrams	mg
<u>VOLUME</u>				
fl oz	fluid ounces (US)	29.57	milliliters	mL
gal	Gallons (liq)	3.785	liters	liters
ft ³	cubic feet	0.0283	meters cubed	m ³
yd ³	cubic yards	0.765	meters cubed	m ³
Note: Volumes greater than 1000 L shall be shown in m ³				
<u>TEMPERATURE</u> (Exact)				
°F	Fahrenheit	5/9 (°F-32)	Celsius	°C
<u>ILLUMINATION</u>				
fc	Foot-candles	10.76	lux	lx
fl	foot-lamberts	3.426	Candela/m ²	Cd/m ²
<u>FORCE and PRESSURE or STRESS</u>				
lbf	pound-force	4.45	newtons	N
psi	pound-force per square inch	6.89	kilopascals	kPa

These factors conform to the requirement of FHWA Order 5190.1A
 *SI is the symbol for the International System of Measurements

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
<u>LENGTH</u>				
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	km kilometers	0.621	Miles (statute)	mi
<u>AREA</u>				
mm ²	millimeters squared	0.0016	square inches	in ²
m ²	meters squared	10.764	square feet	ft ²
m ²	meters squared	1.196	square yards	yd ²
km ²	kilometers squared	0.39	square miles	mi ²
ha	hectares (10,000 m ²)	2.471	acres	ac
<u>MASS (WEIGHT)</u>				
g	grams	0.0353	Ounces	oz
kg	kilograms	2.205	Pounds	lb
mg	megagrams (1000 kg)	1.103	Short tons	T
<u>VOLUME</u>				
mL	milliliters	0.034	fluid ounces	fl oz
liters	liters	0.264	Gallons (liq)	gal
m ³	meters cubed	35.315	cubic feet	ft ³
m ³	meters cubed	1.308	cubic yards	yd ³
<u>TEMPERATURE</u> (Exact)				
°C	Celsius	9/5 (°C+32)	Fahrenheit	°F
<u>ILLUMINATION</u>				
lx	lux	0.0929	Foot-candles	fc
Cd/m ²	Candela/m ²	0.2919	foot-lamberts	fl
<u>FORCE and PRESSURE or STRESS</u>				
N	newtons	0.225	pound-force	lbf
kPa	kilopascals	0.145	pound-force per square inch	psi

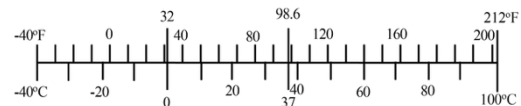


TABLE OF CONTENTS

TABLE OF CONTENTS.....	v
LIST OF TABLES.....	viii
LIST OF FIGURES	ix
ACKNOWLEDGEMENTS.....	xiii
EXECUTIVE SUMMARY	xiv
ES.1 Introduction.....	xiv
ES.2 Problem Statement and Research Objective	xiv
ES.3 Scope of Study and Document Outline	xv
ES.4 Literature Review	xv
<i>ES.4.1 Bridge Deck Evaluation Techniques</i>	<i>xv</i>
<i>ES.4.2 Computer Vision for Bridge Element Inspection.....</i>	<i>xvii</i>
ES.5 Proposed Condition States for Concrete Decks.....	xx
ES.6 Proposed Condition States for Steel Girders	xxi
ES.7 Bridge Deck and Steel Girder Deficiency Datasets	xxiv
ES.8 Three Pipelines for Bridge Deck Condition Assessments.....	xxvii
<i>ES.8.1 Lidar-Based Bridge Deck Map Generation and Delamination Detection</i>	<i>xxviii</i>
<i>ES.8.2 Drone-RGB-Based Bridge Deck Map Generation and Delamination Detection.....</i>	<i>xxxi</i>
<i>ES.8.3 Drone-Thermal-Based Bridge Deck Map Generation and Delamination Detection.....</i>	<i>xxxiii</i>
ES.9 Steel Girder Condition Assessment.....	xxxvi
ES.10 Computer Vision Based Bridge Inspection and Reporting Software	xxxviii
ES.11 Best Practices to Collect Data	xxxix
<i>ES.11.1 Best Practices to Collect Lidar Data.....</i>	<i>xxxix</i>
<i>ES.11.2 Best Practices to Collect RGB and Thermal Data</i>	<i>xl</i>
ES.12 Summary and Conclusions.....	xlvi
CHAPTER 1 – INTRODUCTION.....	1
1.1 Introduction.....	1
1.2 Problem Statement and Research Objective.....	2
1.3 Scope of Study and Document Outline.....	3
CHAPTER 2 – LITERATURE REVIEW	4
2.1 Introduction.....	4
2.2 Bridge Element Inspection and Assessment	4
2.2.1 Bridge Decks.....	4
2.2.2 Superstructure Elements	5
2.2.3 Substructure Elements	7
2.3 Bridge Deck Evaluation Techniques.....	8
2.4 Computer Vision for Bridge Element Inspections.....	12
CHAPTER 3 – QUANTITATIVE CONDITION STATES FOR SELECT BRIDGE ELEMENTS	22
3.1 Introduction.....	22
3.2 AASHTO Condition States for Concrete Decks.....	22
3.3 Proposed Condition States for Concrete Decks	23
3.4 AASHTO Condition States for Steel Girders	24
3.5 Proposed Condition States for Steel Girders.....	26
CHAPTER 4 – BRIDGE DECK AND STEEL GIRDER INSPECTION DATASETS.....	27
4.1 Introduction.....	27
4.2 Existing Image Datasets.....	27
4.3 New Image Datasets for Concrete Bridge Decks.....	28
4.4 New Image Dataset for Bridge Steel Girders.....	31
CHAPTER 5 – RGB BASED AI DELAMINATION DETECTION	32

5.1 Introduction.....	32
5.2 Three Pipelines for Bridge Deck Condition Assessments	32
5.3 Computer Vision Background	33
5.3.1 Deep Convolutional Neural Networks (DCNNs)	33
5.3.2 Semantic Segmentation of Images.....	35
5.3.3 Stereovision and World Mapping	36
5.3.4 Camera Calibration.....	39
5.3.5 Homography Matrix.....	43
5.3.6 Feature Detection	43
5.3.7 Feature Matching.....	47
5.4 Bridge Deck 2D RGB Map Generation	48
5.4.1 Bridge Deck RGB Map Generation using iPhone-Based Lidar Method	48
5.4.2 Bridge Deck RGB Map Generation using Drone-Based RGB Images	50
5.5 AI Based Delamination Detection for Bridge Decks using 2D RGB Maps	50
5.5.1 RGB Data Preparation for Model Training	50
5.5.2 Selected Neural Networks for RGB Data Analysis	52
5.5.3 Neural Network Model Training on RGB Dataset.....	54
5.6 AI Based Delamination Detection on 2D RGB Maps from either Lidar or Drone Data	56
5.7 AI Based Delamination Detection and Quantification in RGB Images.....	56
5.7.1 Detection of Delamination in RGB Images.....	56
5.7.2 Quantification of Delamination Condition States in RGB Images	57
5.8 Results of AI-Based Delamination Detection in RGB Images	59
5.8.1 Local Assessment of Neural Networks for Delamination Using iPhone Based Lidar Images.....	59
5.8.2 Global Assessment of Neural Networks for Delamination Using iPhone Based Lidar Images.....	62
5.8.3 Global Assessment of Neural Networks for Delamination Using Drone Based RGB Images.....	65
CHAPTER 6 – THERMAL BASED AI DELAMINATION DETECTION	67
6.1 Introduction.....	67
6.2 Proposed Methodology to Detect Bridge Deck Delamination in Thermal Images.....	67
6.3 Proposed Methodology to Develop Bridge Deck Thermal Map	68
6.4 Perspective View of Stitched Images.....	76
6.5 AI Based Delamination Detection for Bridge Decks using 2D Thermal Maps	78
6.5.1 Selected Neural Networks for Thermal Data Analysis	78
6.5.2 Normalization Methods for Thermal Images.....	84
6.5.3 Neural Network Model Training on Thermal Dataset.....	84
6.5.4 Performance Indices of Neural Networks on Thermal Images	88
6.5.5 Effects of Different Thermal Data Normalization Methods on Network Performance.....	89
6.5.6 Performance of Neural Networks on Delamination Detection in Thermal Images	90
6.5.7 Samples of Delamination Detections in Thermal Images Using Selected Model	92
6.6 Samples of Delamination Detection on Thermal Maps of Bridge Decks	95
CHAPTER 7 – RGB BASED STEEL GIRDER CONDITION ASSESSMENT WITH A HUMAN-IN- THE-LOOP METHOD	103
7.1 Introduction.....	103
7.2 Image Warping and Scaling.....	103
7.3 Manual Defect Annotation for Steel Girders	104
CHAPTER 8 – COMPUTER AIDED BRIDGE INSPECTION AND REPORTING SOFTWARE	108
8.1 Introduction.....	108
8.2 Software Module for iPhone Based Lidar Inspection of Concrete Bridge Decks	109
8.3 Software Module for Drone Based RGB Inspection of Concrete Bridge Decks	112
8.4 Software Module for Drone Based Thermal Inspection of Concrete Bridge Decks.....	115
8.5 Software Module for Drone Based Inspection of Steel Girders.....	116
8.6 Software Module for Camera Calibration.....	117

CHAPTER 9 – BEST PRACTICES FOR DATA COLLECTION	120
9.1 Introduction.....	120
9.2 Best Practices to Collect Lidar Data	120
9.3 Best Practices to Collect RGB and Thermal Data	120
9.3.1 <i>Flying Height of Drone from Concrete Bridge Decks</i>	120
9.3.2 <i>Gimble Angle</i>	121
9.3.3 <i>Overlap Between Thermal Images</i>	123
9.3.4 <i>Suitable Time Period to Collect Thermal Image Dataset from Bridge Sites</i>	124
CHAPTER 10 – SUMMARY AND CONCLUSIONS	127
10.1 Summary.....	127
10.2 Conclusions.....	128
REFERENCES	130

LIST OF TABLES

Table ES.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)	xx
Table ES.2 – Proposed Condition States for Delamination and Cracking in Concrete Decks	xxi
Table ES.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)	xxii
Table ES.4 – AASHTO Condition States for Steel Protective Coating (AASHTO MBEI, 2019)	xxiii
Table ES.5 – Proposed Condition States for Steel Girders	xxiii
Table ES.6 – Summary of Past Studies on Bridge Related Image Datasets	xxiv
Table ES.7 – Distribution of RGB Image Dataset Used in Neural Networks	xxix
Table ES.8 – Hyperparameters of Selected Deep Learning Models During Training on RGB Image Dataset	xxix
Table ES.9 – Global Assessment of Neural Networks for Delamination Detection of 18 Bridge Decks	xxx
Table ES.10 – Performance Parameters for Different Variants of U-Net Architecture for Thermal Image Data	xxxv
Table 2.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)	5
Table 2.2 – AASHTO Condition States for Prestressed Concrete Girders (AASHTO MBEI, 2019)	6
Table 2.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)	7
Table 3.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)	22
Table 3.2 – Proposed Condition States for Delamination, Spalling, Patching, and Cracking in Concrete Decks	23
Table 3.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)	24
Table 3.4 – AASHTO Condition States for Steel Protective Coating (AASHTO MBEI, 2019)	25
Table 3.5 – Proposed Condition States for Steel Girders	26
Table 4.1 – Summary of Past Studies on Bridge Related Image Datasets	27
Table 5.1 – Camera Calibration Parameters for Devices Used in Present Study	43
Table 5.2 – Bridge Deck RGB Dataset Distribution Used in Neural Network	52
Table 5.3 – Hyperparameters of Evaluated Deep Learning Models During Training on RGB Dataset	54
Table 5.4 – Algorithm to Compute Delaminated Area	57
Table 5.5 – Algorithm to Determine Delamination Condition State and Generate Damage Map	59
Table 5.6 – Local Assessment of Neural Networks for Bridge Deck Delamination Detection in RGB Images	60
Table 5.7 – Global Assessment of Neural Networks for Delamination Detection in RGB Maps of 18 Bridge Decks	63
Table 5.8 – Performance of Neural Networks in Detection of Bridge Deck Delamination in RGB Images	65
Table 6.1. Different Normalization Methods for Thermal Images	84
Table 6.2. Performance of U-Net with Different Normalization Methods for Thermal Data	89
Table 6.3. Performance Parameters for Different Variants of U-Net Architecture	91

LIST OF FIGURES

Figure ES.1 – Data Collection using iPhone’s Lidar and Drone’s Thermal and RGB Cameras	xxv
Figure ES.2 – Sample RGB and Thermal Images with Inspector Annotations	xxvi
Figure ES.3 – Sample RGB Images for Steel Girders	xxvii
Figure ES.4 – Proposed Computer Vision Framework for Delamination Detection in Bridge Decks ..	xxviii
Figure ES.5 – A Sample of PolyCam Generated 3D Mesh Viewed in MeshLab	xxviii
Figure ES.6 – 2D Maps of Bridge Decks Developed from iPhone Point Cloud Data	xxix
Figure ES.7 – AI-based Delamination Detection Flowchart on 2D Maps of Bridge Decks	xxx
Figure ES.8 – Lidar-Based Delamination Assessment of a 75.4-m (248-ft) Bridge	xxxi
Figure ES.9 – 2D Maps of Bridge Decks Developed from Drone’s RGB Images	xxxii
Figure ES.10 – Delamination Assessment of an 89.3-m (293-ft) Bridge Deck Using Drone’s RGB Images	xxxiii
Figure ES.11 – Workflow for Delamination Detection in Concrete Bridge Decks Using Thermal Imaging	xxxiv
Figure ES.12 – 2D Maps of Bridge Decks Developed from Drone’s RGB-Thermal Images	xxxiv
Figure ES.13 – Performance of U-Net in Detection of Delamination on an 89.3 m (293 ft) Bridge Deck	xxxvi
Figure ES.14 – Workflow for Steel Girder Defect Quantification Using a Human-in-the-Loop Method	xxxvii
Figure ES.15 – Sample of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method	xxxviii
Figure ES.16 – Bridge Computer Vision Inspection Tools (Br.CVIT)	xxxix
Figure ES.17 – Data Collection Using iPhone/iPad Lidar Sensor	xl
Figure ES.18 – Flying Height of Drone to Cover Single Lane or Full Width of Bridges	xl
Figure ES.19 – Sample of Drone RGB-Thermal Images Collected at Different Heights from Bridge Decks	xli
Figure ES.20 – Recommended Gimble Angle for RGB and Thermal Imaging	xli
Figure ES.21 – Sample Images with Different Gimble Angles	xlii
Figure ES.22 – Comparison of Stitched Images Based on Data Collected at Slanting Angle and Pointing Downward	xlii
Figure ES.23 – Required Overlap in Successive Drone Imaging	xliii
Figure ES.24 – Suitable Time for Thermal Data Collection	xliv
Figure ES.25 – Delamination Detection by Neural Network for Thermal Images Collected at Different Time of a Summer Day	xl v
Figure 2.1 – NDE Methods Used to Inspect Concrete Bridge Decks	8
Figure 2.2 – Use of Robots to Inspect Bridges	9
Figure 2.3 – Results of Different Thermal Data Analysis Methods for Bridge Decks (Ichi & Dorafshan, 2022)	10
Figure 2.4 – Results of Thermal Data Analysis Using an AI Model (Ichi & Dorafshan, 2024)	11
Figure 2.5 – Current Methods of Generating Thermal Maps for Bridge Decks	12
Figure 2.6 – Deck Damage Detection Framework Using Video and Infrared Imaging (Matsumoto et al., 2014)	13
Figure 2.7 – Computer Vision-based Localization and Mapping of Bridge Deck Cracks (Won & Sim, 2020)	14
Figure 2.8 – 3D Reconstruction and Defect Detection Framework (Perry et al., 2020)	14
Figure 2.9 – Loosened Bolt and Corrosion Detection Framework (Ali et al., 2021)	15
Figure 2.10 – Post-Earthquake RC Column Damage Assessment Framework (Tazarv et al., 2022)	16
Figure 2.11 – GAN-based Transfer Learning Damage Detection Framework (Dunphy et al., 2022)	17
Figure 2.12 – Bolt Detection and Damage Identification Framework (Jiang et al., 2023)	18
Figure 2.13 – Damage Detection and Localization Framework (Jiang et al., 2023)	19

Figure 2.14 – A Damage Localization Technique by Jiang et al. (2023)	19
Figure 2.15 – Crack Detection and Quantification Framework (Kao et al., 2023).....	20
Figure 2.16 – 3D Reconstruction and Crack Detection Framework (Mirzazade et al., 2023).....	21
Figure 4.1 – Data Collection using iPhone’s Lidar and Drone’s Thermal and RGB Cameras.....	29
Figure 4.2 – Sample RGB and Thermal Images from Various Devices Including Inspector Annotations	30
Figure 4.3 – Sample RGB Images for Steel Girders.....	31
Figure 5.1 – Proposed Computer Vision Framework for Delamination Detection of Bridge Decks	33
Figure 5.2 – Depth Estimation from Stereo Camera Setup.....	37
Figure 5.3 – Raw Thermal Image Dataset Collected for Thermal Camera Calibration.....	40
Figure 5.4 – Processed Thermal Image Dataset Collected for Thermal Camera Calibration	41
Figure 5.5 – RGB Image Dataset Collected for RGB Camera Calibration.....	41
Figure 5.6 – Thermal Camera Calibration Steps.....	42
Figure 5.7 – RGB Camera Calibration Steps.....	42
Figure 5.8 – Continued	46
Figure 5.9 – Bridge Deck Feature Detections by SIFT Detector	47
Figure 5.10 – Sample of Feature Matching for a Bridge Deck	48
Figure 5.11 – A Sample of PolyCam Generated 3D Mesh of a Bridge Deck.....	48
Figure 5.12 – 2D Maps of Bridge Decks Developed from Point Cloud (Mesh) Data	49
Figure 5.13 – Flowchart for Bridge Deck 2D Map Extraction from Point Clouds	49
Figure 5.14 – Generation of 2D RGB Map for Entire Bridge Deck.....	50
Figure 5.15 – Samples of Raw and Augmented Data and their Labels Used in Training	51
Figure 5.16 – U-Net Developed by Ronnenberger et al. (2015).....	52
Figure 5.17 – Attention U-Net Framework Developed by Oktay et al. (2018)	53
Figure 5.18 – Recurrent Residual U-Net Framework Developed by Alom et al. (2018)	54
Figure 5.19 – U-Net Validation Score Trend during Training.....	54
Figure 5.20 – Attention U-Net Validation Score Trend during Training.....	55
Figure 5.21 – Recurrent Residual U-Net Validation Score Trend during Training.....	55
Figure 5.22 – Quantification of Condition States for Delamination.....	58
Figure 5.23 – Samples of Local Assessments of Neural Networks for Delamination	61
Figure 5.24 – Delamination Assessment of a 75.4-m (248-ft) Bridge Using Lidar Imaging	63
Figure 5.25 – Delamination Assessment of a 52.7-m (173-ft) Curved Bridge Using Lidar Imaging.....	64
Figure 5.26 – Delamination Assessment of a 52.4-m (172-ft) Skewed Bridge Using Lidar Imaging.....	64
Figure 5.27 – Delamination Assessment of an 89.3-m (293-ft) Bridge Using Drone RGB Imaging	66
Figure 6.1 – Workflow for Delamination Detection in Concrete Bridge Decks Using Thermal Images ...	67
Figure 6.2 – Generation of 2D Thermal Maps for Entire Bridge Decks.....	68
Figure 6.3 – Final Stitched RGB and Thermal Maps for Bridge Shown in Fig. 6.2.....	69
Figure 6.4 – Effects of Drones’ Field of View on Images	69
Figure 6.5 – Stitched Deck RGB and Thermal Maps for Bridge 15215030.....	70
Figure 6.6 – Stitched Deck RGB and Thermal Maps for Bridge 06185230.....	71
Figure 6.7 – Stitched Deck RGB and Thermal Maps for Bridge 03239216.....	72
Figure 6.8 – Stitched Deck RGB and Thermal Maps for Bridge 06158158.....	73
Figure 6.9 – Stitched Deck RGB and Thermal Maps for Bridge 03240216.....	74
Figure 6.10 – Stitched Deck RGB and Thermal Maps for Bridge 03239257.....	75
Figure 6.11 – Stitched Deck RGB and Thermal Maps for Bridge 06185190.....	76
Figure 6.12 – Perspective Corrected RGB and Thermal Maps of a Bridge Deck	77
Figure 6.13 – U-Net Architecture for Thermal Images.....	78
Figure 6.14 – Attention U-Net Architecture (Oktay et al., 2018)	79
Figure 6.15 – Attention Gate used in Attention U-Net (Oktay et al., 2018).....	79
Figure 6.16 – Residual U-Net (ResUnet) Architecture (Zhang et al., 2018)	80
Figure 6.17 – Transformer ResU-Net (TransResU-Net) Architecture (Tomar et al., 2022).....	81
Figure 6.18 – Transformer (Vaswani et al., 2017).....	81

Figure 6.19 – Recurrent Residual U-Net (R2U-Net) Architecture (Alom et al., 2018).....	82
Figure 6.20 – Recurrent Residual Convolution Block used in R2U-Net Architecture (Alom et al., 2018)82	
Figure 6.21 – ResU-Net++ Architecture (Jha et al., 2019)	83
Figure 6.22 – Atrous Spatial Pyramidal Pooling (ASPP) (Chen et al., 2017)	83
Figure 6.23 – Dice Loss of U-Net During Training and Validation on Thermal Dataset.....	85
Figure 6.24 – Dice Loss of Attention U-Net During Training and Validation on Thermal Dataset.....	85
Figure 6.25 – Dice Loss of ResUnet During Training and Validation on Thermal Dataset.....	86
Figure 6.26 – Dice Loss of TransResU-Net During Training and Validation on Thermal Dataset.....	86
Figure 6.27 – Dice Loss of R2U-Net During Training and Validation on Thermal Dataset.....	87
Figure 6.28 – Dice Loss of ResUNet++ During Training and Validation on Thermal Dataset.....	87
Figure 6.29 – Performance of U-Net with Different Normalization Methods for Thermal Images	89
Figure 6.30 – Sample Delamination Detected by U-Net Using Different Normalization Methods for Thermal Images	90
Figure 6.31 – Performance of Different Variants of U-Net Architecture in Detecting Delamination in Thermal Images	91
Figure 6.32 – Sample Delamination Detection by Different U-Net Models in Thermal Images	92
Figure 6.33 – High Performance of Select Neural Network in Detecting Delamination in Thermal Images	93
Figure 6.34 – Low Performance of Select Neural Network in Detecting Delamination in Thermal Images	94
Figure 6.35 – Performance of Select Neural Network in Detecting Delamination in Individual Thermal Images Superimposed on RGB Images	95
Figure 6.36 – Performance of U-Net in Detecting Delamination of an 89.3 m (293 ft) Long Bridge Deck	96
Figure 6.37 – Performance of U-Net in Detecting Delamination of a 77.4 m (253.9 ft) Long Bridge Deck	97
Figure 6.38 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Long Bridge Deck	98
Figure 6.39 – Performance of U-Net in Prediction Delamination on 36.3 m (119.1 ft) Bridge Deck.....	99
Figure 6.40 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Long Bridge Deck	100
Figure 6.41 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Bridge Deck.....	101
Figure 6.42 – Performance of U-Net in Detecting Delamination of a 77.4 m (253.9 ft) Long Bridge Deck	102
Figure 7.1 – Workflow for Steel Girder Defect Quantification Using a Human-in-the-Loop Method	104
Figure 7.2 – Sample 1 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method .	105
Figure 7.3 – Sample 2 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method .	106
Figure 7.4 – Sample 3 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method .	107
Figure 8.1 – Bridge Computer Vision Inspection Tools (Br.CVIT) Landing Page	109
Figure 8.2 – Software Modules.....	109
Figure 8.3 – 3D Map of a Bridge Deck Shown in PolyCam.....	110
Figure 8.4 – Module-1 for iPhone-based Lidar Data	110
Figure 8.5 – Extracted 2D Map from 3D Mesh using Module-1	111
Figure 8.6 – Sample Damage Report Generated by Module-1	111
Figure 8.7 – Overlapping RGB Images of Bridge Deck with Delamination Marks	112
Figure 8.8 – Sample RGB Image of a Bridge Deck with Delamination in Green	112
Figure 8.9 – Stitched 2D RGB Map of a Bridge Deck	113
Figure 8.10 – Sample Damage Report Generated by Module-2	114
Figure 8.11 – Overlapping RGB Images of Bridge Deck.....	115
Figure 8.12 – Overlapping Thermal Images of Bridge Deck.....	115
Figure 8.13 – Stitched RGB and Thermal Maps of a Bridge Deck	115

Figure 8.14 – Sample Damage Report Generated by Module-3	116
Figure 8.15 – RGB Image of Steel Girders.....	116
Figure 8.16 – Corrected Perspective of Steel Girder Images Using Four Known Points	117
Figure 8.17 – Sample Damage Report Generated by Module-4	117
Figure 8.18 – Sample RGB and Thermal Images for Camera Calibration	118
Figure 8.19 – Sample RGB Camera Calibration Parameters Generated by Module-5.....	118
Figure 8.20 – Sample Thermal Camera Calibration Parameters Generated by Module-5.....	119
Figure 9.1 – Data Collection Using iPhone/iPad Lidar Sensor.....	120
Figure 9.2 – Flying Height of Drone to Cover Single Lane and Full Width of Bridges.....	121
Figure 9.3 – Sample of Drone RGB-Thermal Images Collected at Different Heights from Bridge Decks	121
Figure 9.4 – Recommended Gimble Angle for RGB and Thermal Imaging	122
Figure 9.5 – Sample Images with Different Gimble Angles.....	122
Figure 9.6 – Comparison of Stitched Images Based on Data Collected at Slanting Angle and Pointing Downward.....	123
Figure 9.7 – Required Overlap in Successive Drone Imaging.....	123
Figure 9.8 – Suitable Time for Thermal Data Collection	125
Figure 9.9 – Delamination Detection by Neural Network for Thermal Images Collected at Different Time of a Summer Day	126

ACKNOWLEDGEMENTS

The work presented herein was conducted with a joint support from the Alaska Department of Transportation and Public Facilities (Alaska DOT&PF), the National Center for Transportation Infrastructure Durability and Life-Extension (TriDurLE), and the Center for Transformative Infrastructure Preservation and Sustainability (CTIPS); both centers are a member of the University Transportation Center (UTC) funded by the U.S. Department of Transportation (USDOT). The contents of this document reflect the views of the authors, who are responsible for the facts and accuracy of the information presented. This document does not constitute a standard, specification, or regulation.

The support and constructive feedback from the project advisory panel, Jesse Escamilla, Ben Fetterhoff, David McAdoo, Jared Jones, and Elmer Marx of Alaska DOT&PF are greatly appreciated. The authors are indebted to Joshua Olsen of the South Dakota Department of Transportation (SDDOT) for support and technical input while inspecting bridges. The research team is thankful to graduate student Andrew Grismer and undergraduate student Rylan Lipetzky for their assistance and efforts.

EXECUTIVE SUMMARY

ES.1 Introduction

Regular maintenance of transportation infrastructure such as bridges is crucial in extending their service life while reducing long-term repair costs. Transportation agencies such as state Departments of Transportations (DOTs) and the Federal Highway Administration (FHWA) seek emerging technologies and innovative programs to enhance asset management and maintenance. Furthermore, timely inspections of bridges by trained personnel are mandated per national and state specifications.

The Alaska Department of Transportation and Public Facilities (DOT&PF) is responsible for condition assessment of approximately 1,000 bridges in the state. Of which, approximately 44% are in good condition, 49% are in fair condition, and 7% are rated poor based on the 2023 National Bridge Inventory (NBI) data (Infobridge, 2024).

Alaska DOT&PF engineers inspect approximately 500 bridges each year. The Alaska Bridges and Structures Manual (2017) specifies different types of inspections such as inventory, routine, in-depth, and damage. In general, visual inspection is the common practice for the inventory and routine inspections and is combined with other tools such as non-destructive evaluation (NDE) in other inspections for enhanced assessment. The inspector must complete both an NBI inspection (a 10-level condition rating for bridges and bridge components such as decks, superstructures, substructures following the FHWA Specifications for the National Bridge Inventory, 2022) and an element level inspection (a four-level element-by-element condition state assessment based on the AASHTO MBEI, 2019) per bridge.

Nationwide, most steps of bridge inspections such as data collection, defect quantification, and damage reporting are done manually. Furthermore, each of these steps is time consuming, error prone, and hard to repeat in the following inspections due to the manual nature of these activities. For example, the “deck delamination mapping” requires manual detection and measurement of delaminated concrete, patched areas, exposed reinforcing steel bars, and spalling. Such measurements often require traffic control for the safety of inspection crew, and the findings might not be consistent if done by different inspectors.

Computer vision, a field of artificial intelligence (AI) that can analyze scenes at human level performance, can speed up defect identification and quantification only using images of bridge elements. Furthermore, this and other AI tools can be utilized to expedite and unify reporting. Even though the academic and industry level developments are interesting and on the right track for field deployment, they usually lack practicality and are overwhelming for everyday users. For example, the academic-level computer programs are not as user-friendly as commercial software, and the industry-level software packages are so enlarged targeting multiple assets preventing them from being practical. In summary, current tools do not fully address the needs of DOTs for routine bridge inspections.

ES.2 Problem Statement and Research Objective

As discussed above, despite novel inspection procedures and emerging technologies, bridge

inspections in Alaska and nationwide remain predominantly manual. For example, defect detection and quantification such as deck delamination mapping are done manually, which require significant time and labor. Furthermore, existing NDE methods are not common in routine inspection since they are time consuming, costly, and require expert input for data processing. AI-based inspection tools are emerging, but the current tools are either too complex for routine use or not tailored to DOT needs. The main objective of the present study was to develop practical AI tools that help inspectors with defect detections and quantifications, and to facilitate the inspection reporting following the DOT&PF standard requirements.

ES.3 Scope of Study and Document Outline

To achieve the project goals, a few bridge elements (concrete bridge decks and steel girders) were targeted for further investigation, inspection databases including photographs of the selected elements with and without damage were compiled, and computer vision tools were developed for the selected elements to detect the element defects, quantify the defects, and produce a report following the DOT standard practices.

The present report includes a summary of the activities that were focused on the quantification and mapping of concrete bridge deck delamination and steel girder paint loss using Light Detection and Ranging (lidar) sensors of smartphones/tablets, drones equipped with Red-Green-Blue (RGB) and/or thermal cameras, and AI methods. First, a comprehensive literature review of the current scientific trend and AI tools is presented on delamination detection, bridge map generation, and damage state estimation (**Ch. 2**). Then, a new definition of condition states for delamination of concrete bridge decks and steel girder damages suitable for computer programming is proposed in **Ch. 3**. Subsequently, more than 50 bridges were inspected, and a comprehensive dataset of inspector annotated delamination and steel girder damages was compiled (**Ch. 4**). The development of an AI-based inspection software (Br.CVIT) that can detect, quantify, and map the inspector's annotations, and produce a delamination map of whole bridge deck including their condition states is discussed in **Chapters 5-8**. **Chapter 7** provides a human-in-loop method of damage quantification for steel girders. The best practices to collect lidar, RGB and thermal images of bridge decks are discussed in **Ch. 9**. Finally, the summary and conclusions of the study are discussed in **Ch. 10**.

The present executive summary provides an overview of these activities. Detailed discussion can be found in the following chapters of the report.

ES.4 Literature Review

Routine bridge inspection is mandated by the state and federal governments. A brief review of recent trends on bridge inspections using NDE techniques and computer-vision assisted methods is discussed herein.

ES.4.1 Bridge Deck Evaluation Techniques

In current practice, bridges are inspected mostly manually, and their defects are also quantified manually. Visible damage is easy to detect; however, internal defects such as concrete delamination and steel bar corrosion are hard to find. Delamination is an internal damage in concrete caused by insufficient pouring, fatigue, freeze-and-thaw cycles, and/or environmental

conditions. Two simple and cost-effective NDE techniques that are widely used in routine bridge inspections are chain-dragging and sound-hammering methods to differentiate the delaminated and intact regions of concrete. Other NDE methods such as impact echo, ground penetrating radar (GPR), electrical resistivity, and infrared thermography exist but they are not common in routine inspections. The following sections provide a brief review of past works in these NDE methods.

The impact echo method (e.g., Gucunski & Nazarian, 2010) detects air pockets, cavities, or delamination in concrete bridge decks by generating acoustic waves in a frequency range of 5-40 kHz. GPR (e.g., Dinh et al., 2016) uses electromagnetic waves to detect anomalies in concrete, such as delamination and corrosion of reinforcing steel bars. The ultrasonic surface wave method (e.g., Gucunski et al., 2012) detects concrete degradation utilizing two receivers, while the half-cell potential methodology identifies corrosion activity in steel reinforcement.

Infrared thermography is another NDE method (e.g., Washer et al., 2010) employed to detect delamination in concrete structures by analyzing thermal radiation emitted by objects at temperatures above the absolute zero. Thermal cameras detect the heat emitted by the object and the reflective light from the object's surroundings. Thermography sensors may be either cooled or uncooled, functioning in short to medium wavelengths or long wavelengths. Infrared thermography is relatively easy to implement when compared with other NDE methods, especially that drones equipped with both RGB and thermal cameras are currently commercially available with a cost of only a few thousand dollars, which is insignificant compared with other NDE methods and/or inspection equipment. RGB data reveals visible damage, and thermal data can be used to detect subsurface anomalies such as old patching and delamination.

RGB and thermal sensors can be attached to robots or drones to obtain bridge images. La et al. (2015) developed a robotic device which can collect RGB images of bridge decks. The robot can also collect GPR, electric resistivity, impact echo, laser, and GPR data. There was no thermal sensor attached to this robot. Charron et al. (2019) used an unmanned ground vehicle equipped with lidar, GPS, RGB and thermal sensors to scan underside (soffit) of bridge decks. They generated 3D point cloud and colorized the point cloud using RGB and thermal images. They were able to create the RGB and thermal maps of the bridge with reasonable accuracy.

More than ten computational methods have been developed and adopted in the literature to detect delamination in thermal images such as thermal contrast, thermal gradient, adaptive threshold, level-set method, and AI. The thermal contrast method has been widely used in past studies (e.g., Washer et al., 2010; Khan & Bartoli, 2015; Tran et al., 2018; Mac et al., 2019; Vyas et al., 2019; Robison et al., 2020; Kilic & Caner, 2021; Mac et al., 2021; Raja et al., 2021; Rocha et al., 2021; Zheng et al., 2021; and Pedram et al., 2022) to identify delaminated regions in the thermal images of bridge decks. Cheng and Shen (2020) utilized the level-set methodology for delamination detection in concrete specimens, exhibiting greater accuracy than the k-means clustering method. Grayscale values of thermal images were used to detect delaminated regions. Ichi and Dorafshan (2022) utilized an adaptive thresholding technique to detect delamination in concrete bridges, which achieved an accuracy of 69% when a sensitivity analysis for each bridge was performed. They used commercial software to stitch thermal images and used grayscale values of the thermal image for the analysis.

A few studies have tried AI techniques to detect concrete delamination using thermal imaging. For example, Cheng et al. (2020) implemented an encoder-decoder neural network using 1,200 thermal images collected from lab specimens (concrete slabs with artificial delamination) and reported an accuracy of 83.6% on the lab-based test dataset. They further tested their trained model on actual bridge data using grayscale thermal imaging and achieved an accuracy of 51%. Pozzer et al. (2021) compared the performance of VGG16 (Simonyan & Zisserman, 2014) with MobileNetV2 (Sandler et al., 2018) for delamination detection using 517 thermal images with intensity value (grayscale instead of actual temperature) obtained from bridge sites and reported an intersection over union of 59.5% and 59.1% for VGG16 and MobileNetV2, respectively. Ta et al. (2025) utilized well-established deep learning models (e.g., YOLO, Faster RCNN, SSD) to detect delamination in laboratory specimens. They built 12 concrete specimens (each 56×44×7 in.) with artificial delamination by placing nine polystyrene foams in the concrete slabs and obtained 12,000 thermal images after heating the specimens with halogen lamps. Even though they achieved an accuracy of 96%, the AI model was trained only with laboratory images, which may not work on real-world bridge field data. Furthermore, there is a lack of thermal image datasets from actual bridges, as pointed out in SDNET 2021 (Ichi & Dorafshan, 2022), which only includes thermal data of five bridges.

Oh et al. (2012) compared the performance of infrared thermography and impact echo method and found high accuracy for infrared thermography-based inspection when results were compared with the chain-drag method and concrete core samples from bridge decks. They manually stitched the thermal images together by placing the images side by side. Hiasa et al. (2016) mounted thermal cameras on a moving car to capture thermal images. They used three different cooled and uncooled thermal cameras mounted on the vehicle and captured the images from specimens placed alongside the road. They also manually trimmed and combined the thermal images together to obtain a thermal map for the entire bridge deck. Omar et al. (2018) generated thermal map of a bridge deck by manually cropping and then stitching successive thermal images. They did not remove the lens distortions from the thermal images. Thermal images were collected using vehicle mounted thermal cameras. To obtain the delamination map, they employed k-means clustering to segment thermal images.

Overall, literature lacks automated data processing for most of the NDE methods including thermal data. Furthermore, all existing NDE data processing methods heavily rely on the user input and expertise.

ES.4.2 Computer Vision for Bridge Element Inspection

With a widespread use of computer vision techniques, the bridge engineering community is exploring its applications in various tasks including inspections. The following sections provide a brief overview of past studies on the topic with a focus on bridge decks.

Matsumoto (2014) used line sensor camera and infrared thermography to detect bridge deck defects and delamination. Their proposed framework for bridge inspection and repair consists of three phases. In Phase 1, the deck is scanned using video imagery. Phase 2 involves prioritizing the areas of the deck that are distressed, determining their condition state with a scale of 1 to 4, and planning necessary repairs or rehabilitation. In Phase 3, selected spans of the bridge are scanned from underneath to further assess and address the identified issues.

Won and Sim (2020) detected transverse cracks on bridge decks using optical sensors. They used a three-dimensional (3D) computer vision-based localization and mapping technique by utilizing stereovision setup. The proposed framework utilizes multi-view image sequences with a Bag-of-Words representation, eliminating the need for traditional GPS or odometer data for localization. For crack detection, the system uses crack hierarchy and circular histograms instead of conventional methods that rely on line segments and intensities. Additionally, it creates a comprehensive transverse crack dataset based on bespoke ontologies and schemas, covering the entire bridge deck. These advancements aim to enable faster deployment and achieve higher accuracy in large-scale crack mapping and measurement. They collected a dataset of 2,638 image patches, consisting of 1,262 from cracked areas and 1,376 from uncracked (background) areas of bridges. Two classification approaches were applied to these patches including a conventional method and a crack pixel-based representation approach. A support vector machine (SVM) classifier with a radial basis function kernel was trained using the data, with parameters chosen empirically.

Perry et al. (2020) used Structure-from-Motion (SfM) to generate 3D reconstructed model of bridges and to identify target elements. Then they used image processing techniques such as the black hat transform with the Canny edge detection to detect bridge defects. The black hat transform is a morphological operation used in image processing to highlight dark features on a bright background. Specifically, it emphasizes the portions of an image that are smaller than the structuring element used in the transform and darker than their surroundings. The authors also proposed a method to track growth of defects through image comparison.

Ali et al. (2021) used computer vision techniques with a faster region-based convolutional neural network (R-CNN) damage detection framework on data collected using drones for GPS denied locations to identify structural damage related to loosened bolts and steel corrosion. The process begins with identifying the structure to be inspected, followed by deploying an autonomous unmanned aerial vehicle (UAV) equipped with a proportional-integral-derivative (PID) controller for flight stability, an ultrasonic beacon system for precise positioning, and an action camera for capturing visual data. The data is transmitted in real-time using a real-time streaming protocol. A modified faster R-CNN machine learning model, consisting of a region proposal network (RPN), object proposals, and a Fast R-CNN, is employed for image analysis. The model identifies and highlights structural damage. The ground station processes and analyzes the collected data, completing the inspection process. This integration of robotics, computer vision, and machine learning enables efficient and safe autonomous inspection of structures.

Tazarv et al. (2022) used computer vision techniques for inspection and condition assessment of reinforced concrete (RC) bridge columns after earthquakes to automatically determine their conditions. They used deep learning based cascaded detection of deficiencies in their framework using Mask R-CNN. The cascaded detection framework begins with Stage 1, which detects columns in images. If columns are not detected, the user is prompted to find them manually. Once columns are identified, the process moves to Stage 2 for detecting longitudinal bars, followed by Stage 3 for detecting transverse bars. If both bars are detected, the system reports damage state 4 or 5. Stage 4 involves spalling detection; if spalling is found, it results in damage state 3 or 4. If not, it continues to Stage 5 for crack detection, leading to damage state 1 or 2. Their results indicate high performance across all components.

Dunphy et al. (2022) developed a multiclass damage detection framework for concrete structures using a combination of generative adversarial networks (GANs) and CNNs. It begins with a multiclass image database of various concrete conditions, such as undamaged, cracked, construction joint, pitting, and spalling. The database was split into labeled and unlabeled datasets. GAN training was performed on unlabeled data, producing a trained generator and discriminator network. The trained discriminator was then used for transfer learning to initialize the weights and biases of CNN. This CNN was further trained using the labeled data. Finally, hyper-training was applied to the fully connected layer of CNN to optimize it for image classification tasks, leading to accurate identification of different types of concrete damage.

Jiang et al. (2023) developed a framework using object detection models to identify steel bolts and nuts, as well as their conditions, including corrosion and tightness. The proposed method for automatic bolt inspection was a vision-based system leveraging image processing and deep learning algorithms with images captured by UAV. It consisted of three main components: bolt image acquisition, data preprocessing, and bolt damage identification. The image acquisition phase involved using a modified UAV to capture images of bolts from various angles, including the bridge's sides, bottom, and cable clamps. Data preprocessing addresses two key issues: image motion blur due to the UAV's high speed and variations in bolt pixel size caused by differing object distances. To tackle these, an image motion deblurring method using inverse filtering and optical flow was employed to correct motion blur, and an adaptive scale segmentation method with multi-scale template matching and enhanced super-resolution generative adversarial networks (ESRGANs) ensured uniform bolt image sizes of 240×240 pixels. Finally, a two-stage bolt damage identification process was used to detect and classify bolt conditions.

Jiang et al. (2023) proposed a three-stage framework for bridge condition assessment and damage localization using UAV collected data. The authors used the YOLOv3 object detection network to detect several types of damage such as cracks, corrosion, and spalling. During data collection, the authors also used visual simultaneous localization and mapping (VSLAM) techniques to localize each damage on bridges by retrieving camera pose with respect to reference world coordinates. To achieve this, the authors used stereo-inertial SLAM that used VSLAM with Inertial Measurement Unit (IMU) sensor fusion to increase the robustness of the system. Integration of IMU with VSLAM made the process more robust because of a better estimation of the camera poses for each image. Their framework was tested on a long-span river bridge, offering a more rigorous evaluation environment than standard indoor tests due to factors like strong winds and a large detection area.

Kao et al. (2023) proposed a framework to quantify cracks in bridges using UAV images and deep learning-based object detection models. They used planar markers to retrieve scales of images then cracks. To retrieve the crack boundaries, they used a local thresholding method to create a binarized image. Following that, the authors employed a crack width measurement technique in reference to the planar marker. This model performed well in identifying cracks even in images with uneven lighting and complex backgrounds. The study also demonstrated that the overall crack measurement accuracy exceeded 0.22 mm. Two edge detection methods evaluated in the study showed similar performance; however, the Canny edge detector's results

varied with different thresholds, leading to greater discrepancies between measured and actual crack widths. In contrast, the morphological edge detector, which does not rely on thresholds, provided crack edges more consistent with the true dimensions.

Mirzazade et al. (2023) used a 3D reconstruction SfM approach to detect and segment bridge damages based on U-Net, a popular deep learning model for semantic segmentation. SfM is a computer vision technique that reconstructs 3D structures from 2D image (only width and height) sequences taken from different viewpoints. It estimates camera positions and the 3D coordinates of scene points by analyzing the motion and structure captured in the overlapping images. They also collected Light Detection and Ranging (lidar) scans of structures to verify the integrity of the 3D reconstruction using SfM.

Overall, the literature indicates that computer vision is a viable technique to detect and quantify many of the bridge defects using a variety of imaging methods such as RGB, thermal, and lidar. These capabilities can be further expanded when paired with drone-based data collection, which improves coverage, efficiency, and access to hard-to-reach components.

ES.5 Proposed Condition States for Concrete Decks

Delamination is a state of separation of layers within concrete. This may lead to loss of structural integrity of bridge decks and increase their maintenance cost. Detection and forecasting the growth of the delamination is inherently difficult because of the lack of visual perception. AASHTO MBEI (2019) defines four condition states for delamination and other defects of bridge decks as presented in **Table ES.1**.

Table ES.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination /Spall /Patched Area (1080)	None.	Delaminated. Spall 1 in. or less deep or 6 in. or less in diameter. Patched area that is sound.	Spall greater than 1 in. deep or greater than 6 in. diameter. Patched area that is unsound or showing distress. Does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Exposed Rebar (1090)	None.	Present without measurable section loss.	Present with measurable section loss but does not warrant structural review.	
Efflorescence/ Rust Staining (1120)	None.	Surface white without build-up or leaching without rust staining.	Heavy build-up with rust staining.	
Cracking (RC) (1130)	Insignificant cracks or moderate width cracks that have been sealed.	Unsealed moderate width cracks or unsealed moderate pattern (map) cracking.	Wide cracks or heavy pattern (map) cracking.	
Abrasion/ Wear (PSC/RC) (1190)	No abrasion or wearing.	Abrasion or wearing has exposed coarse aggregate but the aggregate remains secure in the concrete.	Coarse aggregate is loose or has popped out of the concrete matrix due to abrasion or wear.	

It is clear that the AASHTO definitions of condition states for deck defects are mostly qualitative. Nevertheless, any defects that are expected to be found using computer tools need to

be quantitative but not qualitative. To this end, **Table ES.2** presents alternative definitions of condition states for concrete decks specifically for delamination, spalling, and patching. The new definitions are measurable and programmable. Furthermore, to minimize variations from AASHTO, the proposed definitions are kept at four levels. Very small delaminated regions are usually hard to find in practice, thus a limit of 7 in² (45 cm²) was proposed for CS-1, which is indeed insignificant compared with the area of the bridge deck. The proposed area for CS-2 follows that of the AASHTO manual at this level. Based on the inspection of more than 50 bridges with experienced inspectors, the research team noticed that a 12 in. (30 cm) by 12 in. (30 cm) area was the smallest region inspectors annotated as delamination. With a 30% reduction, the research team proposed a cap of 110 in² (710 cm²) in CS-3. Any area greater than that is marked as CS-4. It is understood that these limits are subjective; however, the user of the inspection software (the product of this study, **Ch. 8**) can adjust them as needed. Compared with the AASHTO definitions, some parameters, such as the depth of defects, were removed in the new definitions since they are challenging to estimate using computer vision techniques.

Table ES.2 – Proposed Condition States for Delamination and Cracking in Concrete Decks				
Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination, Spall, or Patched Area (1080)	Defected area is less than or equal to 7 in ² (45 cm ²)	Defected area is greater than 7 in ² (45 cm ²) but less than 30 in ² (195 cm ²)	Defected area is greater than 30 in ² (195 cm ²) but less than 110 in ² (710 cm ²)	Defected area is greater than 110 in ² (710 cm ²)
Cracking (RC) (1130)	Crack length is less than 10 in. (25.4 cm)	Crack length is greater than 10 in. (25.4 cm) but less than 20 in. (50.8 cm)	Crack length is greater than 20 in. (50.8 cm) but less than 40 in. (101.6 cm)	Crack length is greater than 40 in. (101.6 cm)

ES.6 Proposed Condition States for Steel Girders

Similar to the deck damages, the AASHTO definitions of condition states for steel girders are also qualitative. For example, **Table ES.3** presents different damage types seen for steel elements and **Table ES.4** presents the condition states for the steel protective coating. Alternative and measurable definitions are required, which can be used in computer programming.

Table ES.5 presents the proposed damage classification for steel girders. In this study, a defect with an area equal to or smaller than 0.3% of the total surface area of the girder was selected as CS-1. For CS-2, the area of defects has a range of 0.3 to 3% of the girder total surface area. For CS-3, the area of defects has a range of 3 to 16% of the girder total surface area. If the area of the defect is more than 16% of the girder total surface area, then it falls under the category of CS-4.

These proposed condition states for steel girders were selected based on a report by Bowman et al. (2022). Note that the crack in steel girders is very critical. A small crack can lead to a catastrophic failure. A fracture critical member inspection may be carried out to inspect the critical girders, and appropriate measures should be taken to ensure safety of steel bridges. Ohio DOT (2014) recommends a CS of 2 for crack lengths up to 2 in., and Wisconsin DOT (2024) proposes a CS of 4 for crack lengths greater than 3 in.

Table ES.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Corrosion (1000)	None.	Freckled rust. Corrosion of the steel has initiated.	Section loss is evident, or pack rust is present but does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Cracking (1010)	None.	Crack that has self-arrested or has been arrested with effective arrest holes, doubling plates, or similar.	Identified crack that is not arrested but does not warrant structural review.	
Connection (1020)	Connection is in place and functioning as intended.	Loose fasteners or pack rust without distortion is present but the connection is in place and functioning as intended.	Missing bolts, rivets, or fasteners; broken welds; or pack rust with distortion but does not warrant a structural review.	
Distortion (1900)	None.	Distortion not requiring mitigation or mitigated distortion.	Distortion that requires mitigation that has not been addressed but does not warrant structural review.	
Settlement (4000)	None.	Exists within tolerable limits or arrested with no observed structural distress.	Exceeds tolerable limits but does not warrant structural review.	
Scour (6000)	None.	Exists within tolerable limits or has been arrested with effective countermeasures.	Exceeds tolerable limits but is less than the critical limits determined by scour evaluation and does not warrant structural review.	The element has impact damage. The specific damage caused by the impact has been captured in CS 4 under the appropriate material defect entry.
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in CS 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in CS 3 under the appropriate material defect entry.	

Table ES.4 – AASHTO Condition States for Steel Protective Coating (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Chalking (Steel Protective Coatings) (3410)	None.	Surface dulling.	Loss of pigment.	Not applicable.
Peeling/Bubbling/Cracking (Steel Protective Coatings) (3420)	None.	Finish coats only.	Finish and primer coats.	Exposure of bare metal.
Oxide Film Degradation Color/Texture Adherence (Steel Protective Coatings) (3430)	Yellow-orange or light brown for early development. Chocolate-brown to purple-brown for fully developed. Tightly adhered, capable of withstanding hammering or vigorous wire brushing.	Granular texture.	Small flakes, less than 1/2-in. diameter.	Dark black color. Large flakes, 1/2-in. diameter or greater, or laminar sheets or nodules
Effectiveness (Steel Protective Coatings) (3440)	Fully effective.	Substantially effective.	Limited effectiveness.	Failed; no protection of the underlying metal.
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 3 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 4 under the appropriate material defect entry.

Table ES.5 – Proposed Condition States for Steel Girders

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Corrosion/Paint loss (1000)	Corrosion/paint loss less than 0.3% of total surface area of the girder	Corrosion/paint loss greater than 0.3% and less than 3%	Corrosion/paint loss greater than 3% and less than 16%	Corrosion/paint loss greater than 16%
Cracking (1010)	Crack length less than 0.25 in	Crack length greater than 0.25 in. but less than 2 in.	Crack length greater than 2 in. but less than 3 in.	Crack length greater than 3 in.
Chalking (Steel Protective Coatings) (3410)	Chalking less than 0.3% of total surface area of the girder	Chalking loss greater than 0.3% and less than 3%	Chalking greater than 3% and less than 16%	Chalking greater than 16%
Peeling/Bubbling (Steel Protective Coatings) (3420)	Peeling/Bubbling less than 0.3% of total surface area of the girder	Peeling/Bubbling greater than 0.3% and less than 3%	Peeling/Bubbling greater than 3% and less than 16%	Peeling/Bubbling greater than 16%

ES.7 Bridge Deck and Steel Girder Deficiency Datasets

Large numbers of images are usually needed for successful training of neural networks. **Table ES.6** presents a summary of past studies that developed image datasets specific to bridge element damage assessment. For example, the CrackForest dataset by Shi et al (2016) includes annotated images of cracks on various concrete structures, including bridges. A derivative of the Common Objects in Context (COCO) dataset by Bianchi et al. (2020) was tailored for bridge inspection tasks, which includes annotated images of bridges highlighting various structural elements and potential defects. Dorafshan et al. (2021) provided a dataset containing images of concrete surfaces, including bridges, annotated for different types of defects such as cracks and spalling.

Table ES.6 – Summary of Past Studies on Bridge Related Image Datasets

Reference	Dataset	Categories	Data Type	Size	Application
Shi et al. (2016)	CrackForest	Cracks	RGB Image	10,000	Detect and analyze cracks on the road surface
Zhang et al. (2016)	CCIC	Cracks	RGB Image	40,000	Detect cracks on concrete surfaces
Rubio et al. (2019)	--	Exposed Rebar, Delamination	RGB Images	734	Detect and quantify exposed rebars and delamination
Zou et al. (2019)	DeepCrack	Cracks	RGB Image	35,100	Detect fine cracks on various road surfaces
Bianchi et al. (2020)	COCO Bridge Damage	Bearing, Cover-plates, Gusset-plates, Stiffeners	RGB Image	774	Detect bearing, cover-plate, gusset-plate, stiffeners and evaluate their conditions
Dorafshan et al. (2021)	SDNET2021	Delamination	Infrared Thermography, Impact Echo, Ground Penetrating Radar	1,936	Detect delamination and reinforcement corrosion
Arya et al. (2022)	RDD2022	Cracks and Potholes	RGB Image	47,420	Detect and classify road damage types, such as cracks, potholes, and surface wear
Tazarv et al. (2022)	RC Column Damage	Cracks, Spalling, Exposed Rebars	RGB Image	3,036	Detect post-earthquake damages of RC bridge columns and determine their damage states

To collect new data for bridge decks, the research team accompanied by inspectors from the South Dakota Department of Transportation (SDDOT) inspected more than 50 bridges in South Dakota in the summers of 2023 and 2024. The inspectors used the chain-drag method, a nondestructive technique, to evaluate bridge decks for delamination and annotated them with paints. The research team then collected RGB images using digital single-lens reflex (DSLR) cameras, iPhone 14 Pro, and DJI drones (Matrice 200, and Mavic 3T) before and after the annotations. Furthermore, PolyCam, a mobile application utilizing iPhone's lidar camera, was used to scan bridge decks to record 3D meshes (point cloud data) of each deck.

RGB images taken by iPhone 14 Pro had a resolution of 8064 by 6048 pixels. Further, the RGB images obtained by the DJI Mavic 3T and the DSLR camera had a resolution of 4000 by 3000 and 4608 by 3456 pixels, respectively. The images taken by the drones were taken approximately at a height of 10 ft (3 meters) from the deck surface. For the images taken by the other cameras, the height varied between 3 to 6 ft (1 to 2 meters). **Figure ES.1** shows the data collection for iPhone-based lidar, and drone based RGB and thermal data. **Figure ES.2** shows a few sample

images from each of these devices. More than 13,000 RGB and 2,000 thermal images were gathered from 50 bridges. Furthermore, the deck of 18 bridges with delamination were lidar scanned and data was processed per bridge. Additional inspection data was also provided by Alaska DOT&PF and was used in the AI model training.

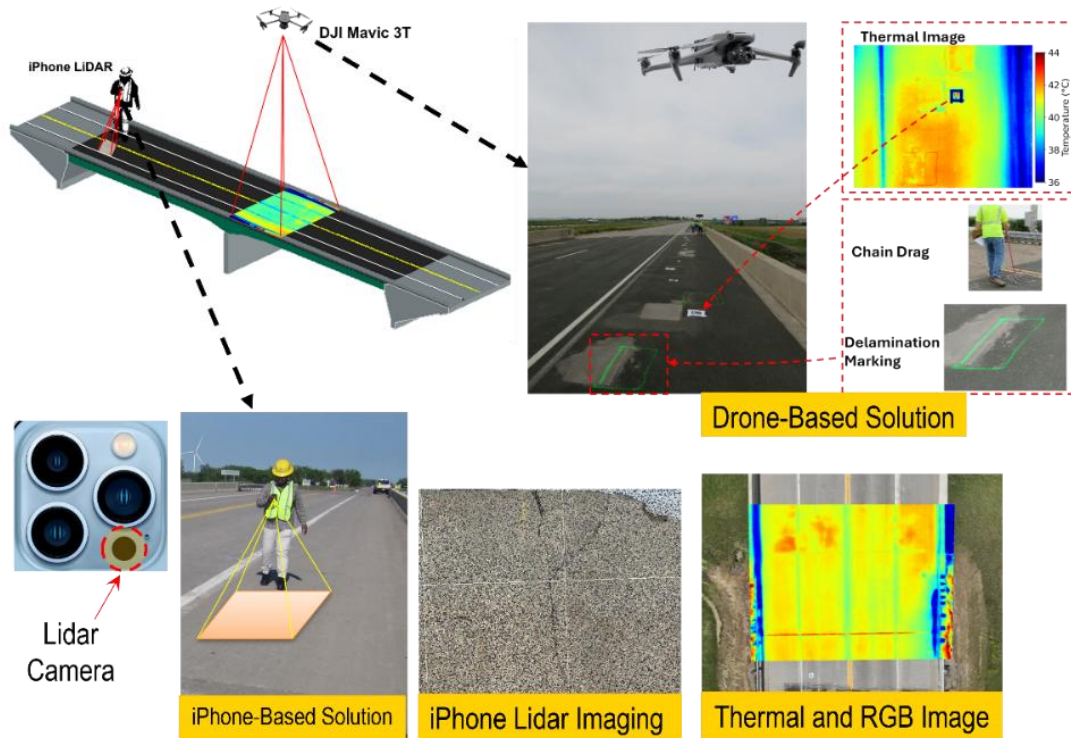
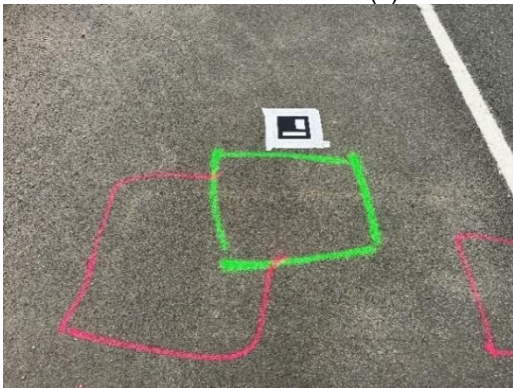


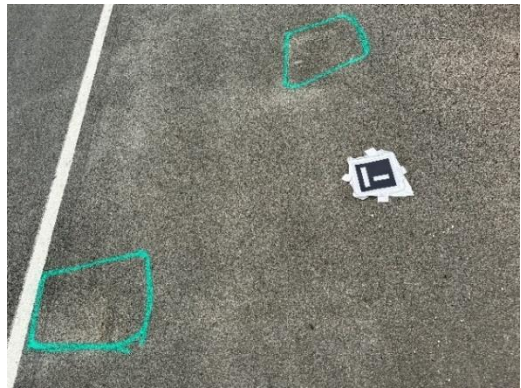
Figure ES.1 – Data Collection using iPhone’s Lidar and Drone’s Thermal and RGB Cameras



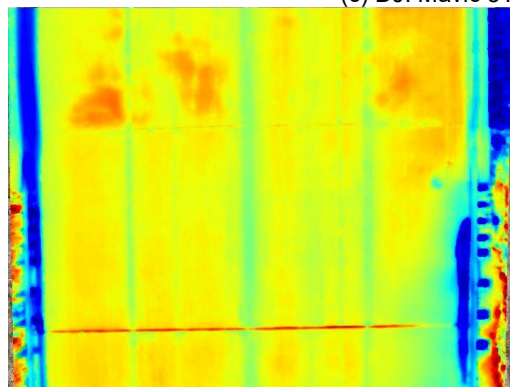
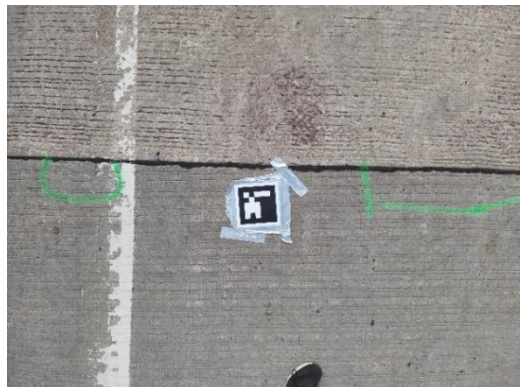
(a) DSLR Camera



(b) iPhone 14 Pro



(c) DJI Mavic 3T (RGB)



(d) DJI Mavic 3T (Thermal)

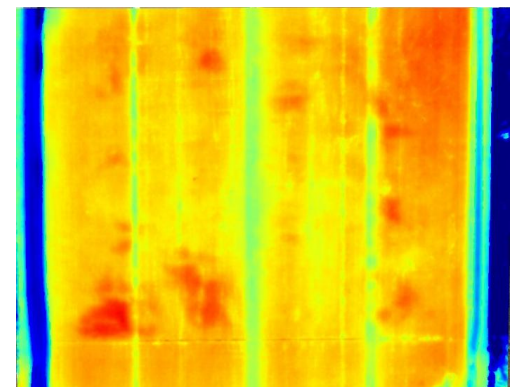


Figure ES.2 – Sample RGB and Thermal Images with Inspector Annotations

For steel girders, images were collected using handheld cameras and drones (e.g., **Fig. ES.3**). More than 6,700 images of steel girders, including corrosion and paint loss, were collected.



Figure ES.3 – Sample RGB Images for Steel Girders

ES.8 Three Pipelines for Bridge Deck Condition Assessments

In the present project, the research team developed a computer vision-based bridge defect detection and condition state estimation software that was mainly focused on delamination in concrete bridge decks (**Chapters 5-8**). Furthermore, the software estimates the paint loss in steel girders using human-in-the-loop method, which is discussed later in **Sec. ES.9** (also **Ch. 7**).

Figure ES.4 shows a high-level architecture of the software for bridge decks, which has several interconnected modules to allow data collection and analyses using different devices and methods. The three main pipelines for bridge decks include lidar-based inspection, RGB-based inspection, and thermal-based inspection.

In the lidar-based method, 3D point-cloud maps of bridge decks are obtained using iPhone's/iPad's lidar camera. The point-cloud data is then run through the software to generate a 2D map of each bridge deck. The software then detects local defects (delamination) using a trained neural network and pinpoints the damage on the global map of the bridge deck.

In the RGB-based method, RGB images can be acquired using different sources (DSLR, smartphone/tablet's cameras, or cameras on UAVs). However, a drone-based data collection is recommended and built-in to develop a full 2D map of bridge decks. Subsequently, a deep learning model trained on a large image dataset is used to detect inspector annotated delamination.

The third pipeline was developed to generate 2D thermal maps of bridge decks using drone-based thermal imaging then another deep learning model detected the deck delamination in thermal maps.

The findings of all pipelines are then quantified based on the proposed damage state definitions discussed in **Sec. ES.5 & 6** (as default values, user can change them) and a report is generated with standard outputs and formats. In the following sections, the components of the software related to bridge decks are briefly discussed.

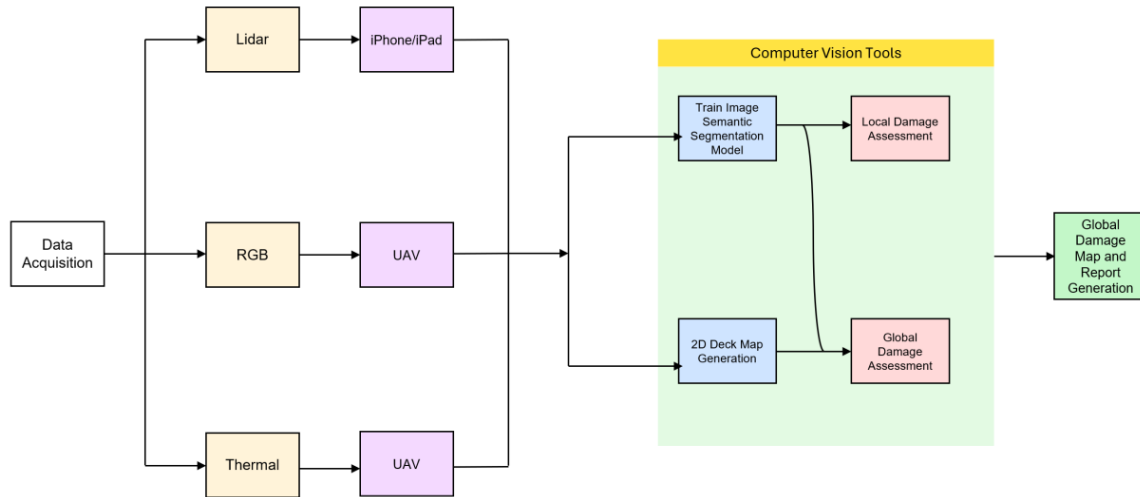


Figure ES.4 – Proposed Computer Vision Framework for Delamination Detection in Bridge Decks

ES.8.1 Lidar-Based Bridge Deck Map Generation and Delamination Detection

Bridge deck maps are generated from 3D point clouds recorded by an iPhone/iPad and an application that uses the device lidar sensor. In the present study, PolyCam was used to collect the point cloud data. At each bridge site, the bridge is scanned lane-by-lane to produce a global 3D map of the bridge deck (e.g., **Fig. ES.5**).

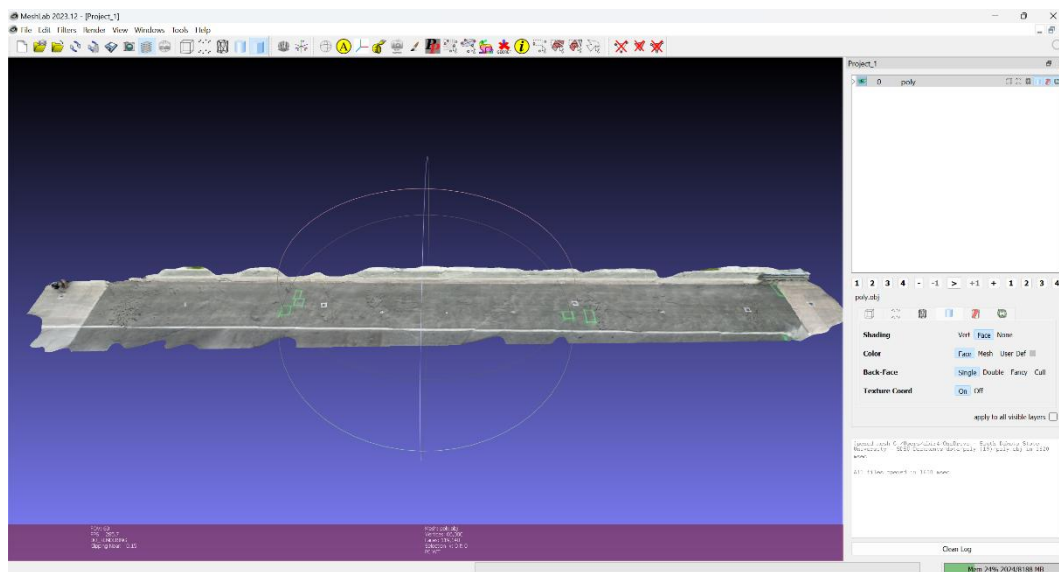


Figure ES.5 – A Sample of PolyCam Generated 3D Mesh Viewed in MeshLab

Subsequently, the 3D scan recorder with an extension of OBJ including its texture map is processed to obtain a 2D image of the bridge deck by mapping all 3D points to a reference plane, which is the bridge deck surface in the present study. **Figure ES.6** shows a sample 2D map of a bridge deck extracted from its 3D map (**Fig. ES.5**).

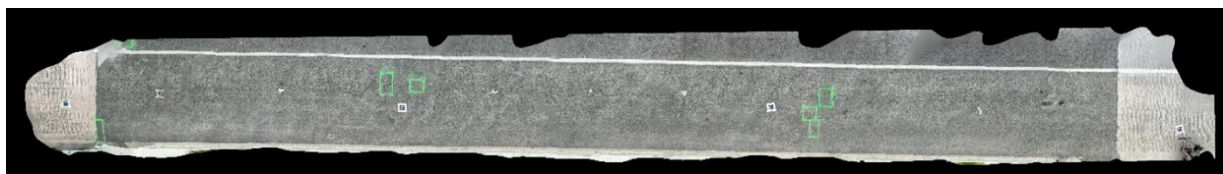


Figure ES.6 – 2D Maps of Bridge Decks Developed from iPhone Point Cloud Data

Once a 2D map of the bridge (either lane by lane or whole width of the bridge) is obtained, it is forwarded to the neural network for semantic segmentation and quantification of the inspector annotations. A large dataset was needed to train deep learning models that can detect inspector annotations (e.g., the green markups in **Fig. ES.6**). Note that the image and lidar dataset compiled in the present study was discussed in the previous section (**Sec. ES.7**). Nevertheless, a data augmentation method was performed to further increase the number of images for training and testing.

In the present study, the image training set to detect markups on bridge decks was color augmented in the Hue, Saturation, and Value (HSV) color space. This augmentation method was applied randomly with a probability of 50%. Furthermore, different geometric augmentation methods such as scaling and rotations were also applied. The scaling ranged from 0.5 to 1 of the training sample with a probability of 50%. A rotation of -45 to +45 degrees was applied to increase the number of samples. The dataset was then split into a training and testing set with a ratio of 90% and 10%, respectively. The training set was further split into a training and validation set of 90% and 10%, respectively. **Table ES.7** presents the sample size for the inspector annotation image dataset during training and testing. Note that the training set of 1,553 was increased to 70,000 after performing the above-mentioned augmentations.

Table ES.7 – Distribution of RGB Image Dataset Used in Neural Networks

Set	Number of Samples
Training	1553
Validation	172
Testing	192

Three different neural networks (U-Net, Attention U-Net, Recurrent Residual U-Net) were selected with different hyperparameters, but the same loss function (the Dice function) was used in all models. **Table ES.8** lists the best hyperparameters of the models during training on the dataset.

Table ES.8 – Hyperparameters of Selected Deep Learning Models During Training on RGB Image Dataset

Model	Batch Size	Learning Rate	Weight Decay	Momentum
U-Net	5	0.0001	1×10^{-8}	0.999
Attention U-Net	2	0.0005	1×10^{-8}	0.99
Recurrent Residual U-Net	4	0.0001	1×10^{-8}	0.99

The model learns from the prepared data to classify each pixel and compares its predictions with benchmark data, which is called a ground truth. The outcome of the prediction determines how much the model has learned or needs to be updated and enhanced. **Figure. ES.7** shows the proposed workflow for delamination detection of concrete bridge decks using inspector markups and its quantification process.

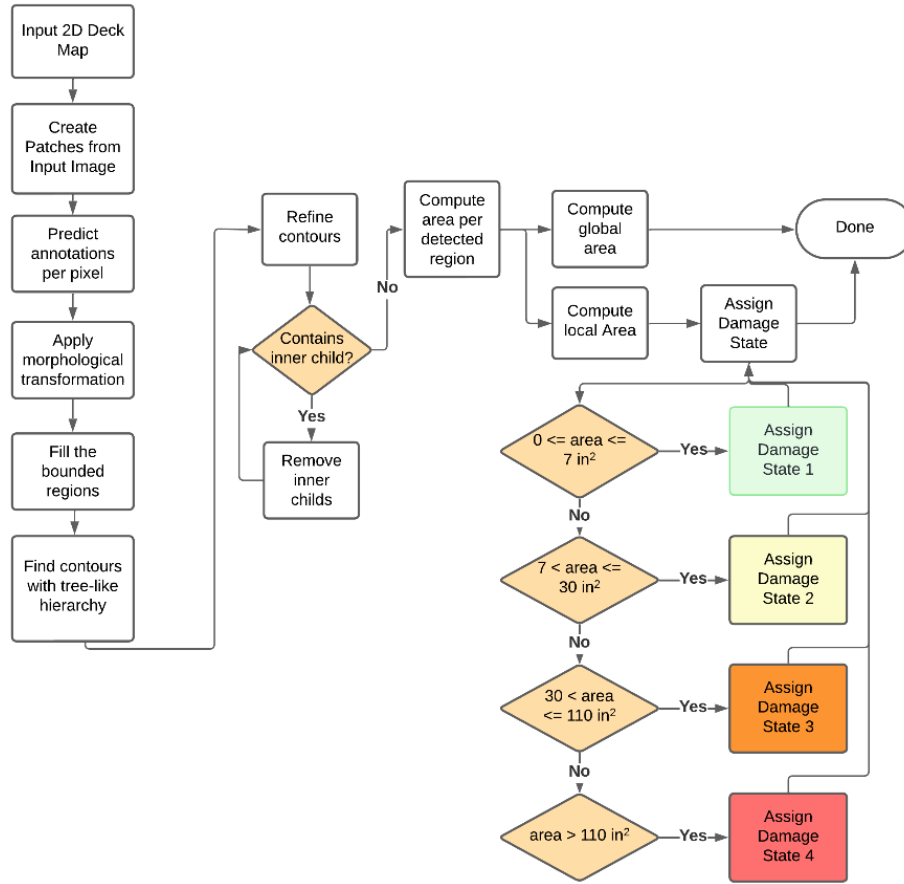


Figure ES.7 – AI-based Delamination Detection Flowchart on 2D Maps of Bridge Decks

A delamination analysis at the global level (e.g., area-to-area at the full length of the bridge) provides a complete assessment of the entire processes and methods of autonomous delamination detection in real applications. **Figure ES.8** shows a sample of lidar-based delamination assessment for a bridge on its full length, which was 248 ft (75.4 meters). **Figure ES.8a** shows the input image, a 2D map of the bridge deck including the marked delamination by the inspector (the enclosed areas with green paint), to the neural network. This input image is then processed to extract patches with a size of 512-by-512 pixels with a stride of 256. This makes each patch have a 50% overlap with the adjacent ones. **Figures ES.8b-d** show the predicted pixels by the three neural networks as the delamination boundaries (finding the green markups automatically). The findings are then used to compute the delamination condition state for individual zone using a contour analysis on the post-processed prediction map. The predicted delaminated pixels are then connected using morphological transformations and passed through contour analysis. After the completion of the contour analysis, which was done to enclose each delamination and to obtain its shape, each of the delaminated areas is then color coded based on the severity of the delamination. The scale information was retrieved from the mesh data.

Table ES.9 presents a summary of global (area to area) evaluation results. The U-Net and Attention U-Net models performed well on the test set while Recurrent Residual U-Net showed

an inferior performance on the test and deck map sets. Overall, U-Net performed better than the other two models and is recommended for field applications.

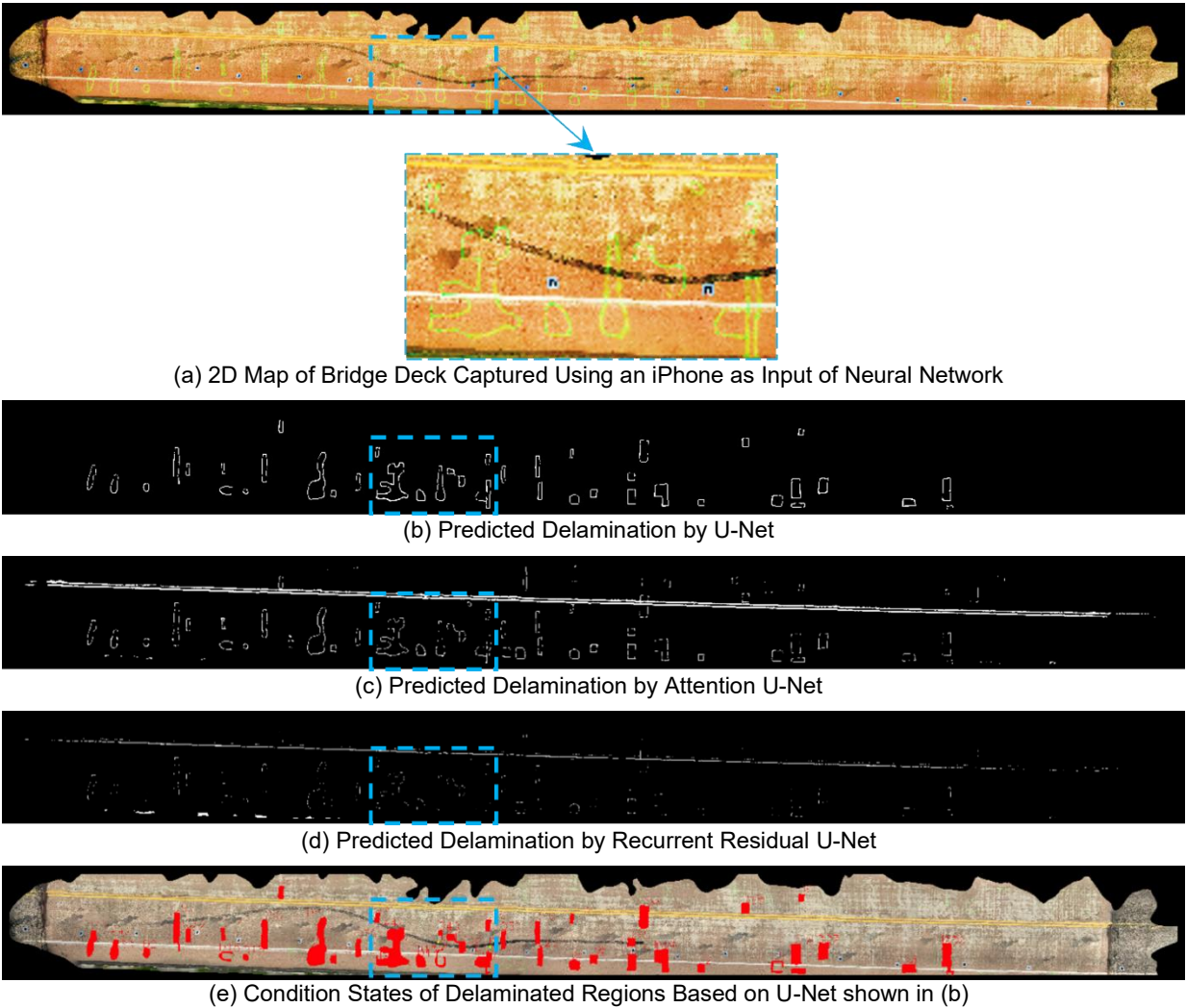


Figure ES.8 – Lidar-Based Delamination Assessment of a 75.4-m (248-ft) Bridge

Table ES.9 – Global Assessment of Neural Networks for Delamination Detection of 18 Bridge Decks	
Model	Deck Map Performance (Area)
U-Net	99.5%
Attention U-Net	84.5%
Recurrent Residual U-Net	< 5%

ES.8.2 Drone-RGB-Based Bridge Deck Map Generation and Delamination Detection

Drone’s RGB images can be used to develop a 2D map of whole bridge deck, then a deep learning model, the same as the one discussed in the previous section, detects the deck delamination. Note that high-elevation images of a bridge can cover its entire deck area; however, the resolution of such images is usually low for a successful AI analysis. By lowering

the height of the drone, a high-resolution image can be captured but may not cover the whole deck. The present study proposes an image stitching method that overcomes this shortcoming by providing a high-resolution image of the entire bridge deck suitable for AI damage detections.

Figure ES.9 shows the proposed image stitching process, which uses camera parameters, feature detection, and feature matching, to develop a 2D map of the entire bridge deck. In this method, RGB images are first rectified using camera calibration parameters. The rectified RGB images are then used to detect features within each image followed by the feature matching for corresponding points in the common regions of two overlapping images. Subsequently, a homography matrix (H_{RGB}) is calculated using these matched feature points. The process is performed on all images and then a 2D map of the bridge deck is formed by image warping and blending. **Figure ES.9** shows an example of such process for 11 overlapping RGB images taken by a drone. Note that the 2D map of the bridge deck in **Fig. ES.9** was shown in a perspective view for clarity.

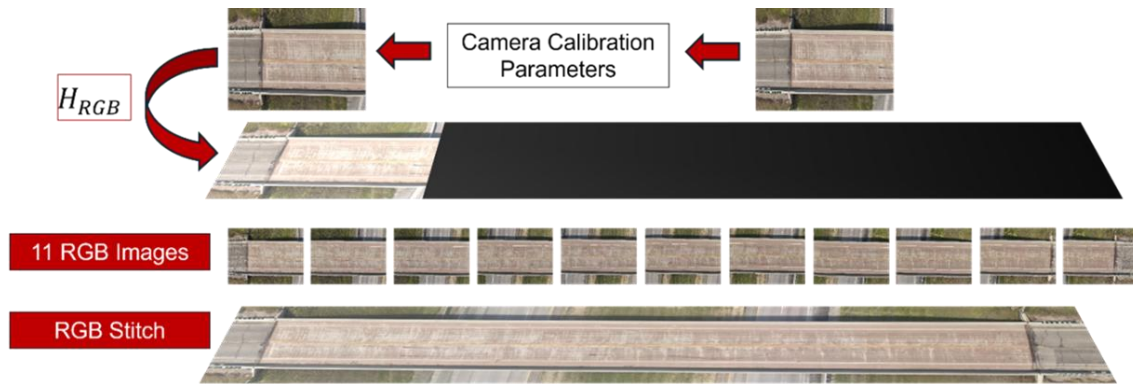


Figure ES.9 – 2D Maps of Bridge Decks Developed from Drone's RGB Images

Figure ES.10 shows a stitched image of a 293 ft (89.3 m) long bridge deck using 50 overlapping images following the above-mentioned method. The delaminated regions were paint marked in green on the bridge deck. Following the training and validation of the deep learning model discussed in **Sec. ES.8.1**, the 2D deck map of this bridge was fed to the AI model to detect the delaminated regions. The neural network successfully detected the paint markings in the image (**Fig. ES.10b**). The accuracy of the neural network to detect the delaminated area was 96%. **Figure ES.10c** shows a superimposed AI-detected delamination map on the bridge deck, which indicates a successful localization and quantification of this defect.

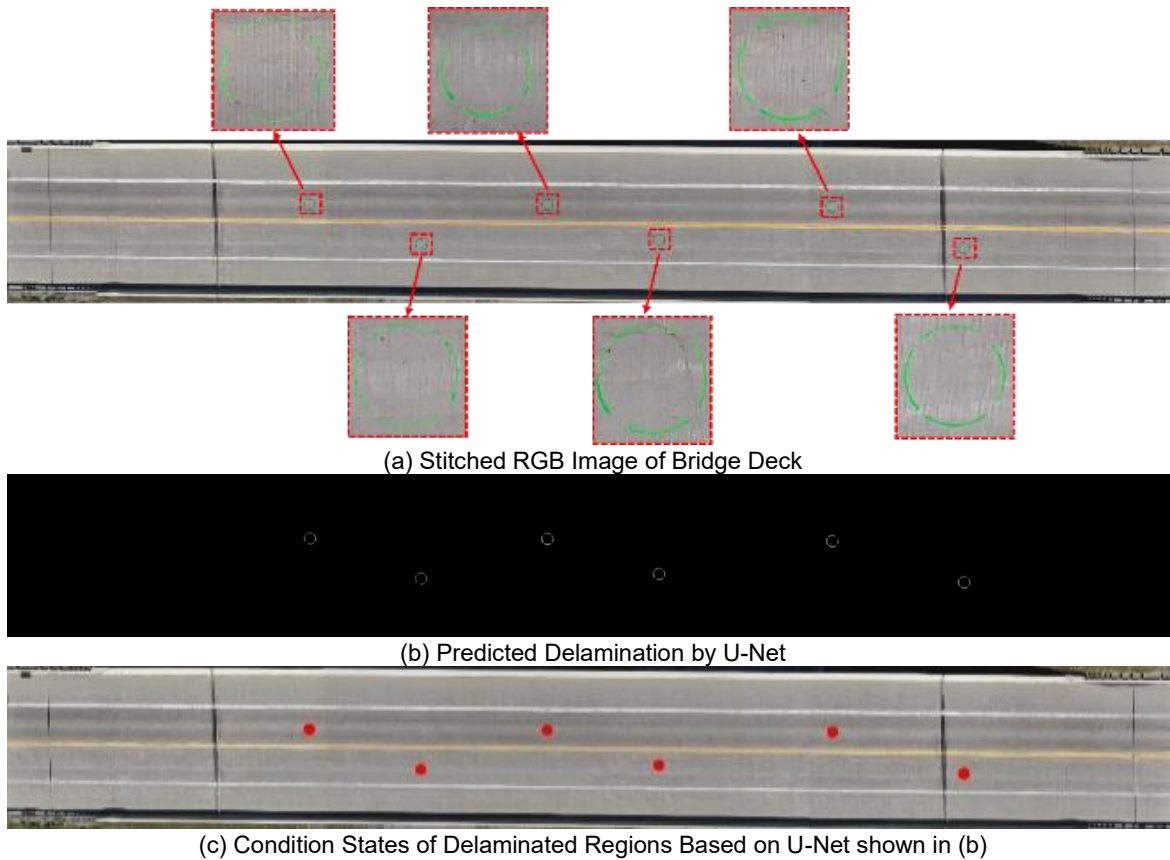


Figure ES.10 – Delamination Assessment of an 89.3-m (293-ft) Bridge Deck Using Drone's RGB Images

ES.8.3 Drone-Thermal-Based Bridge Deck Map Generation and Delamination Detection

Literature lacks systematic methods for the map generations of bridge decks using either RGB or thermal imaging. As discussed in the previous section, novel RGB-based map generation and delamination detection for bridge decks were developed in the present study. Furthermore, the research team developed a novel method of generating thermal maps for bridge decks and trained deep learning models that can automatically detect concrete delamination in thermal images. The thermal map generation and delamination detection for concrete bridge decks are briefly discussed herein.

Figure ES.11 shows the proposed methodology to automatically generate a delamination map for concrete bridge decks using a fused RGB-thermal imaging and computer vision. The steps include collection of overlapping images using a drone that is equipped with a pair of thermal and RGB cameras, which are calibrated and correlated. Subsequently, RGB images are stitched to develop a 2D map of the bridge deck following the method discussed in the previous section. Then, thermal data are superimposed over the RGB data using homography mapping. Subsequently, the fused data is processed to obtain the thermal map of the whole bridge deck (**Fig. ES.12**). The final stage is to analysis the thermal map using a trained neural network to automatically detect delaminated regions (**Fig. ES.13**). Note that the 2D RGB-thermal maps of the bridge deck in **Fig. ES.12** were shown in a perspective view for clarity.

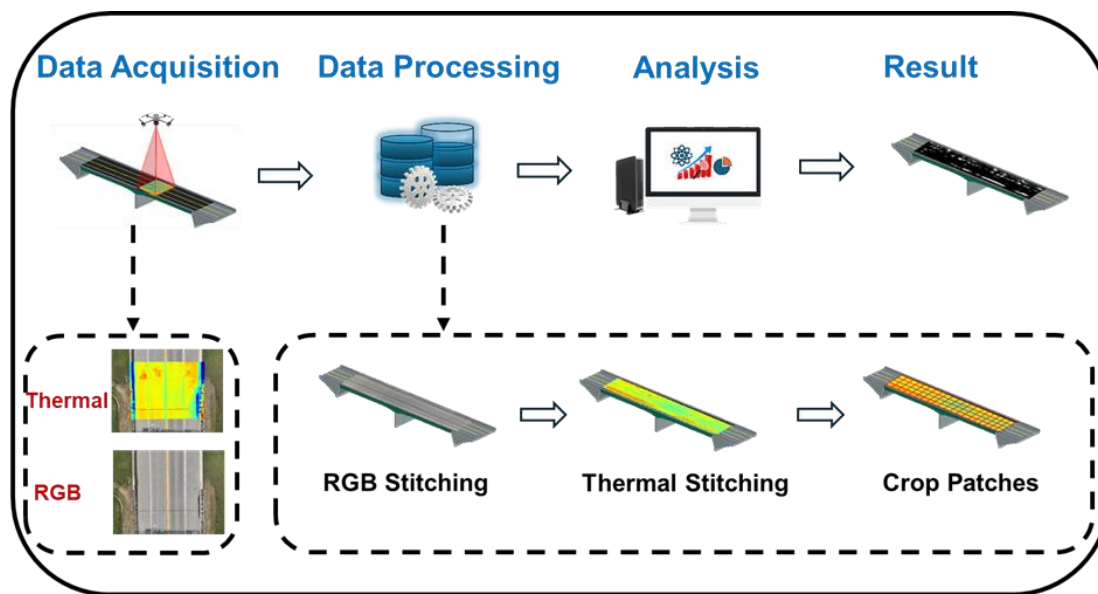


Figure ES.11 – Workflow for Delamination Detection in Concrete Bridge Decks Using Thermal Imaging

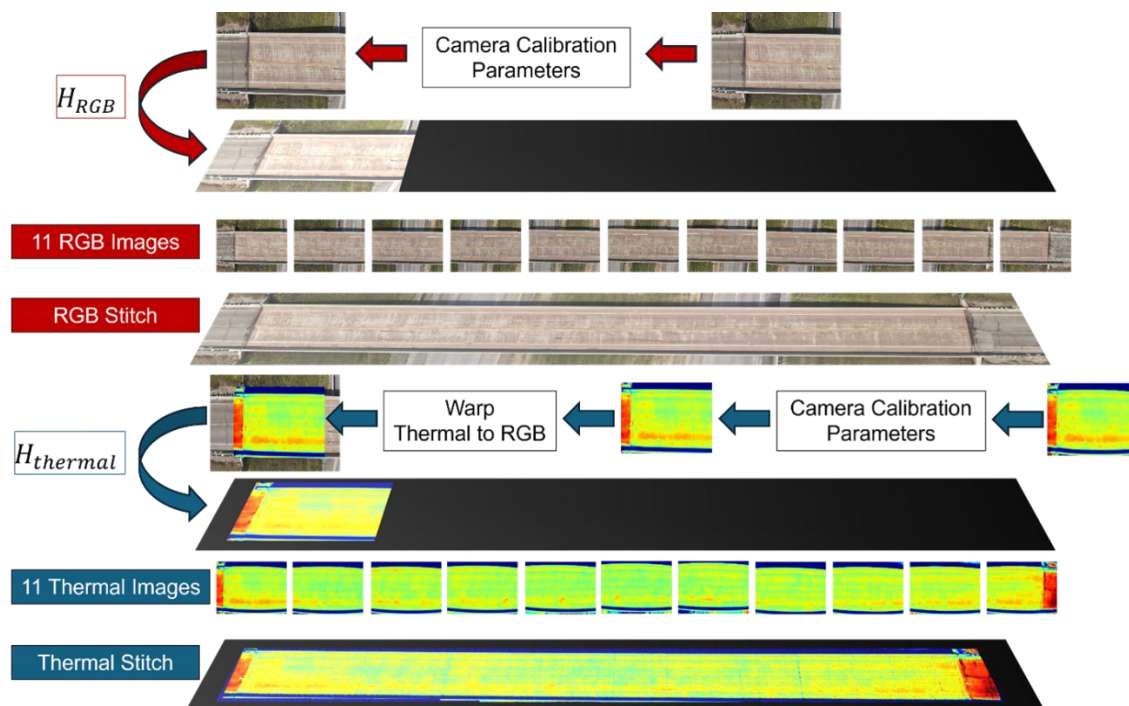


Figure ES.12 – 2D Maps of Bridge Decks Developed from Drone's RGB-Thermal Images

Out of 5,456 original and augmented thermal images that were compiled in the present study, 80% of the image dataset was used to train the models and 20% was used to test the performance of the neural networks. **Table ES.10** presents a summary of the performance of U-Net, and its five different variants used in the present study for thermal image processing. Note that these performance parameters were based on 419 validation images. It can be seen that the U-Net model outperformed the other models in terms of performance parameters, in which it showed an accuracy of 86.2%, a precision of 69.7%, a recall of 76%, an F1-score of 67.4%, and a Jaccard index of 54.5%.

Table ES.10 – Performance Parameters for Different Variants of U-Net Architecture for Thermal Image Data

Parameter	U-Net (%)	Attention U-Net (%)	ResUnet (%)	TransResU-Net (%)	R2U-Net (%)	ResUNet++ (%)
Accuracy	86.2	85.9	78.8	83.6	11.2	85.8
Precision	69.7	60.1	10.6	44.4	2.6	60.8
Recall	75.8	56.3	47.6	30.9	1.1	55.1
F1 score	67.4	58.1	17.3	34.1	1.5	57.6
Jaccard index	54.5	54.1	10.3	24.3	0.8	53.5

Figure ES.13 shows a sample result in which the white color represents the detected delaminated regions of the bridge deck. The figure also includes a corresponding ground truth map for this bridge, which was obtained from the chain-drag method by an expert inspector. Compared with the ground truth, the model showed an accuracy of 77% on the intersection over union. As a global performance assessment, it was found that the total physical delaminated area (e.g., m²) of the prediction was 99% of that of the ground truth. Note that the thermal image showed higher temperatures at both ends of the bridge. This was because vaulted (mostly hollow) abutments were used at the ends. Overall, the neural network was reasonably accurate and was able to detect delamination, their location and areas on the entire bridge deck.

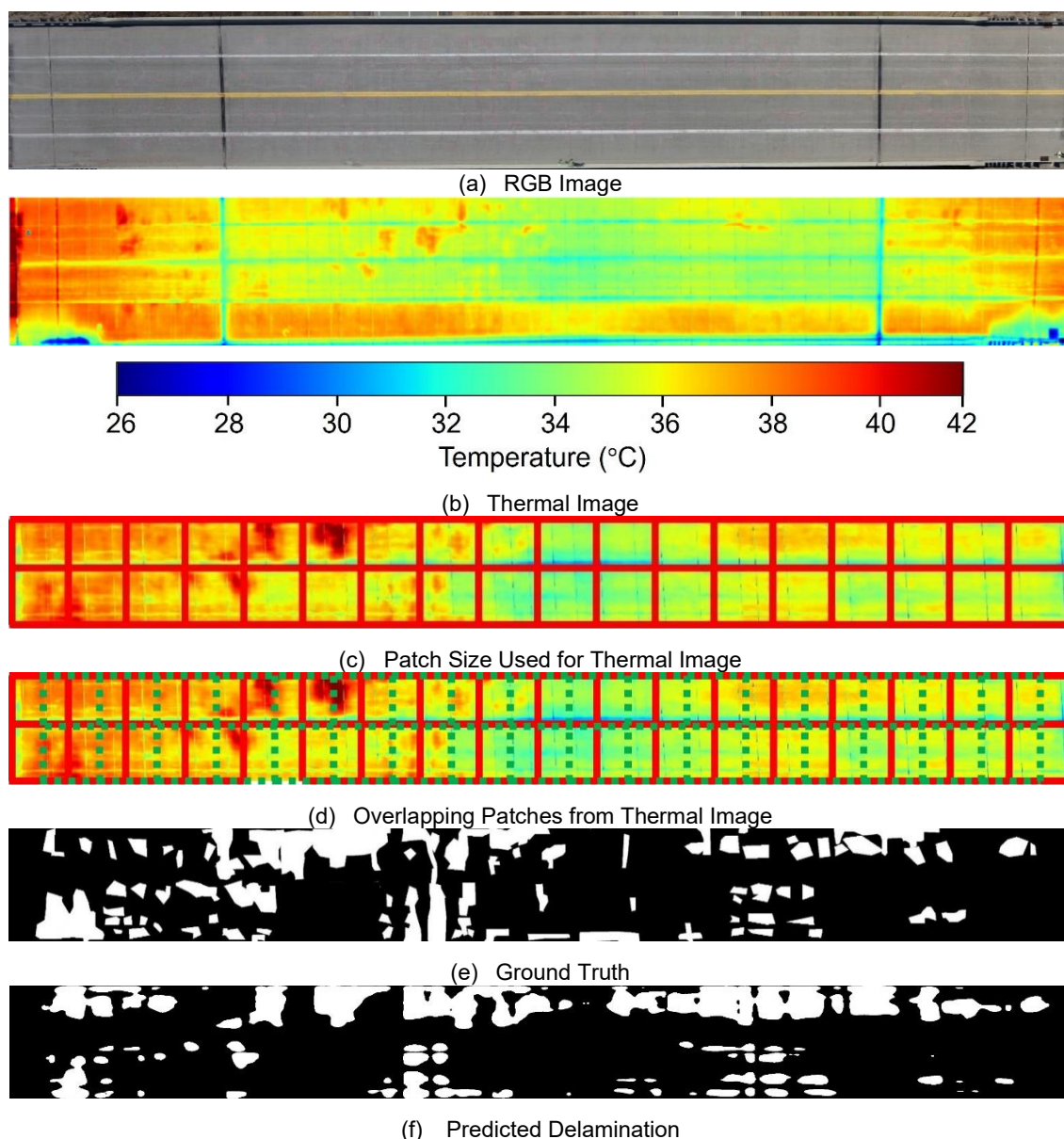


Figure ES.13 – Performance of U-Net in Detection of Delamination on an 89.3 m (293 ft) Bridge Deck

ES.9 Steel Girder Condition Assessment

The present study also offers a human-in-the-loop method for condition assessments of steel girders. **Figure ES.14** shows a flowchart for such assessment. In the proposed method, RGB cameras of drones or conventional devices can be used to capture the images. Camera calibration parameters are then used to undistort and correct each RGB image. Furthermore, in most cases, it is not feasible to capture images of bridge girders at a 90° angle (perpendicular to the girder's web) thus an alternative method is needed to virtually correct the perspective of the image. The study proposes a perspective correction in which the inspector/user selects four points on the girder, which their locations (coordinates on the girder elevation view including relative distances between the points or girder depth) are known. These dimensions can be

collected in the field during bridge inspections or from the structural drawings. The top and bottom of (two) steel girder web stiffeners may be used as those known points (e.g., **Fig. ES.15**). These four points and their coordinates are then used to perform perspective correction of the girder image and to retrieve the scale of the image (the pixel-to-physical-length information). Subsequently, the inspector/user annotates the defected regions of the steel girder. Finally, a damage map will be generated. **Figure ES.15** shows an example of steel girder assessed using this approach. The total defected area (paint loss in this example) was estimated to be 0.23 m^2 (2.5 ft^2) for this girder. Note that the depth of the girder was 0.92 m (3 ft). Based on the proposed damage states for steel girders (**Sec. ES.6**), the software estimates the condition state of each damage.

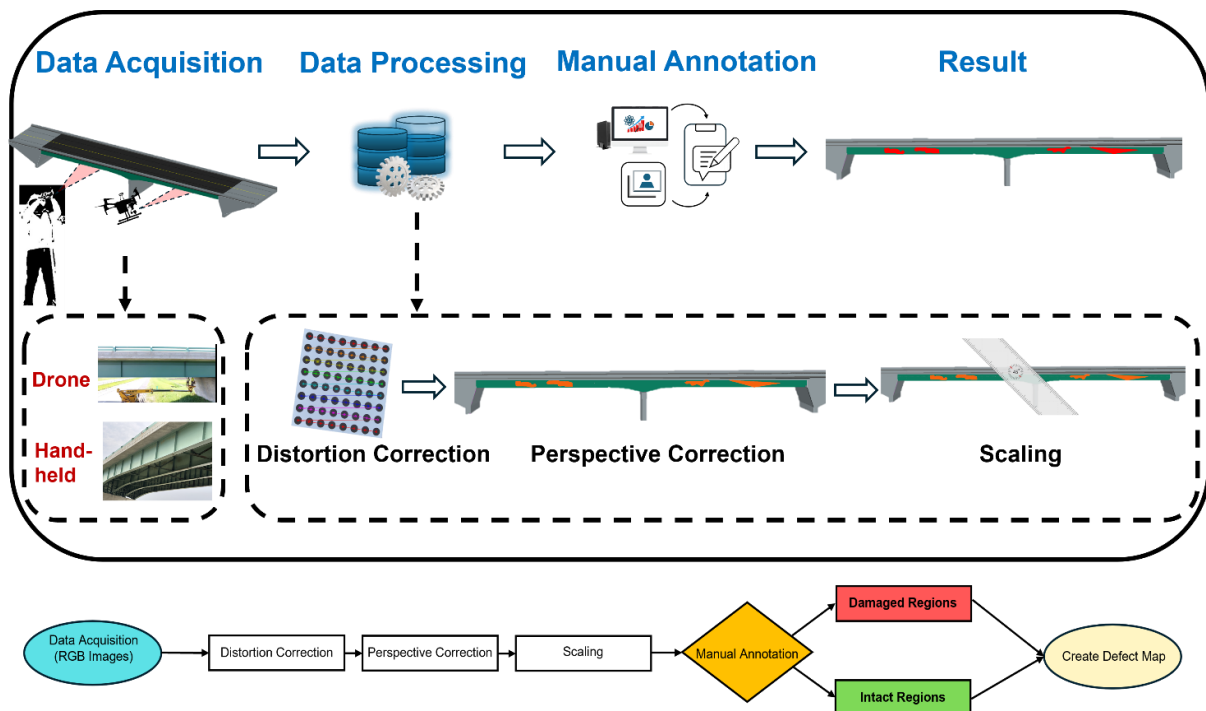


Figure ES.14 – Workflow for Steel Girder Defect Quantification Using a Human-in-the-Loop Method

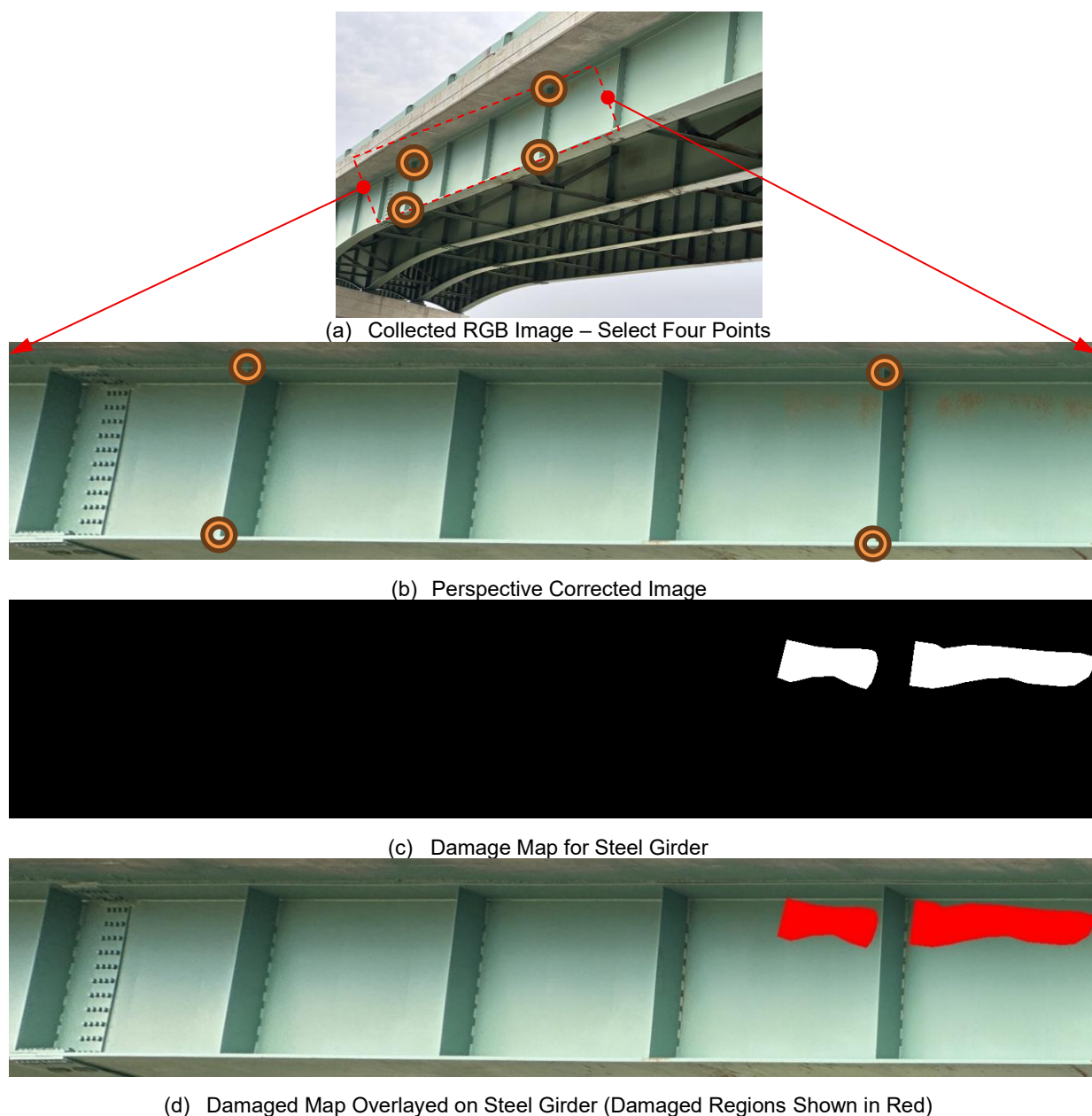


Figure ES.15 – Sample of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method

ES.10 Computer Vision Based Bridge Inspection and Reporting Software

In the present study, the research team proposed three methods to detect defects of concrete bridge decks and to generate inspection reports using lidar data, drone based RGB imaging, and drone-based thermal imaging. Furthermore, a method to estimate steel girder defects was also developed. All these methods, in which computer vision techniques were incorporated, were discussed in the previous sections. A user-friendly software, Bridge Computer Vision Inspection Tools (Br.CVIT), was developed by the research team implementing all the above-mentioned methods and approaches. The software (a screenshot is shown in **Fig. ES. 16**) includes five modules to perform: (1) assessment of bridge decks using lidar data, (2) assessment of bridge

decks using drone's RGB images, (3) assessment of bridge decks using drone's RGB-thermal images, (4) assessment of steel girders using single RGB image, and (5) calibration of RGB and thermal cameras. A comprehensive user guide was developed providing step-by-step procedures to perform each assessment.

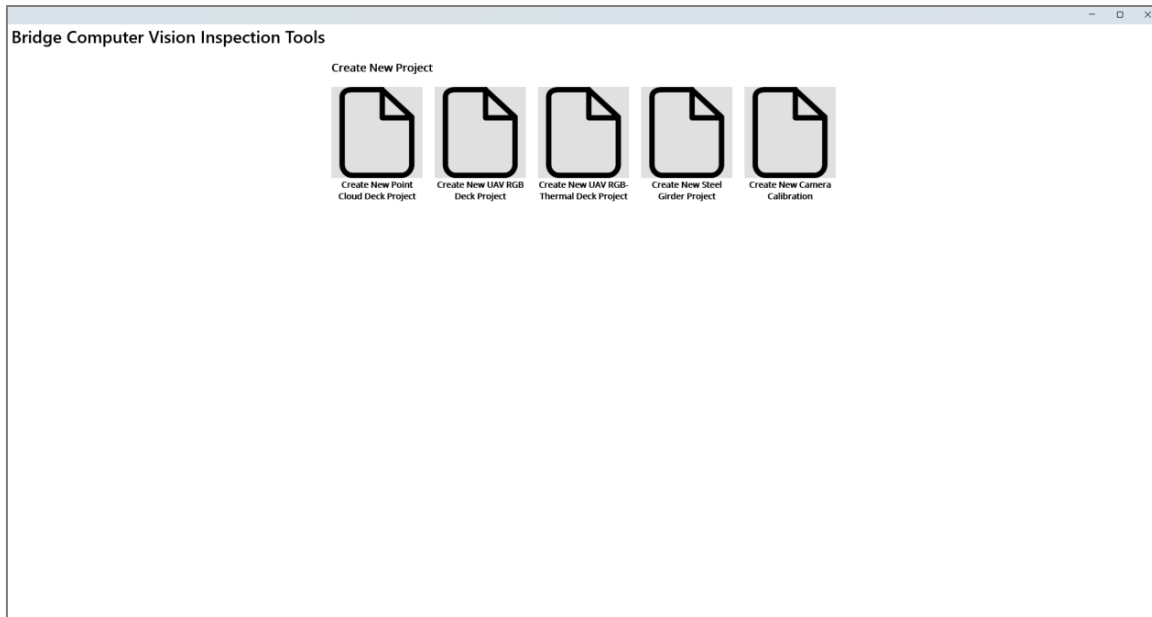


Figure ES.16 – Bridge Computer Vision Inspection Tools (Br.CVIT)

ES.11 Best Practices to Collect Data

During the course of the project, the research team exercised different data collection methods and practices, and found some of those, especially for lidar and thermal data, crucial for successful post-processing by the software. This section provides guidelines on the best practices to obtain inspection data including iPhone/iPad lidar imaging, drone RGB imaging, and drone thermal imaging.

ES.11.1 Best Practices to Collect Lidar Data

To obtain lidar data using iPhone/iPad, PolyCam App, one of many applications available for such data, can be used. The lidar data can then be exported to the software (Br.CVIT) using the Object format. The research team recommends the following as the best practices when collecting the lidar data:

- Scan the bridge deck at height of 5 ft or less.
- Scan one lane at a time.
- Scan along the sunlight to avoid glare.

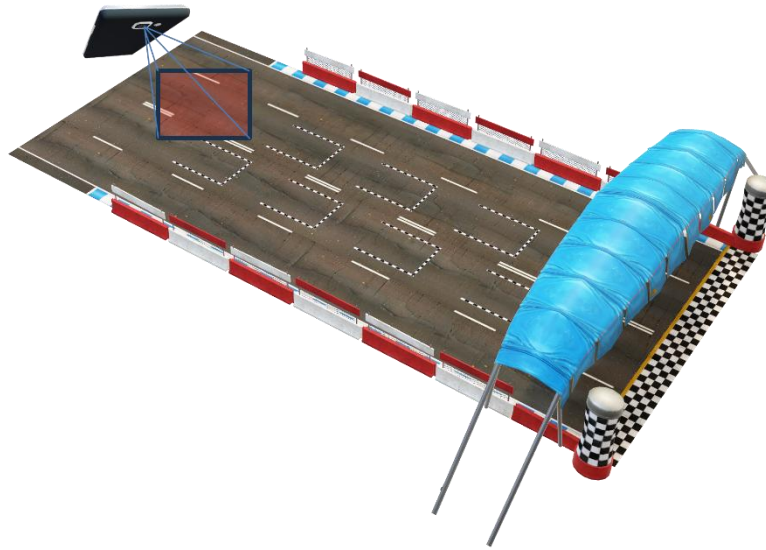


Figure ES.17 – Data Collection Using iPhone/iPad Lidar Sensor

ES.11.2 Best Practices to Collect RGB and Thermal Data

Based on the resolution of RGB and thermal sensors, the drone height must be adjusted for a preferred dimension that can cover a lane, multiple lanes, or the full width of the bridge. **Figure ES.18** shows two cases for single lane and full width of bridges, and **Fig. ES.19** shows sample images taken by a DJI Mavic 3T drone.

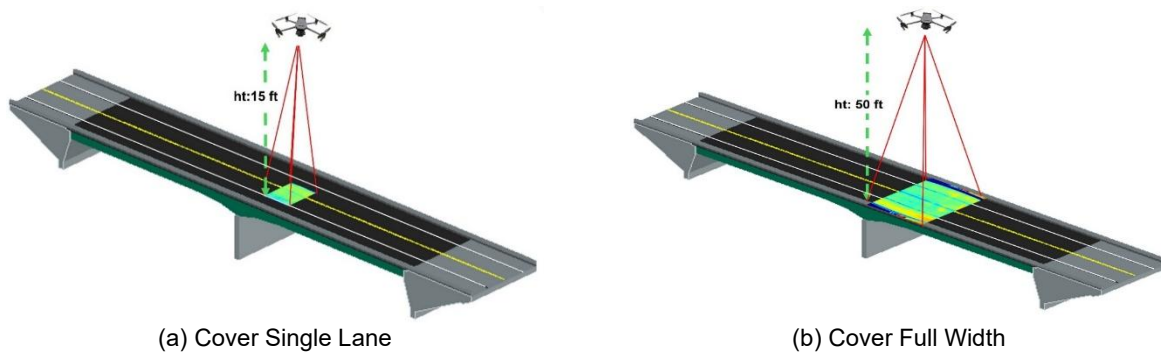


Figure ES.18 – Flying Height of Drone to Cover Single Lane or Full Width of Bridges

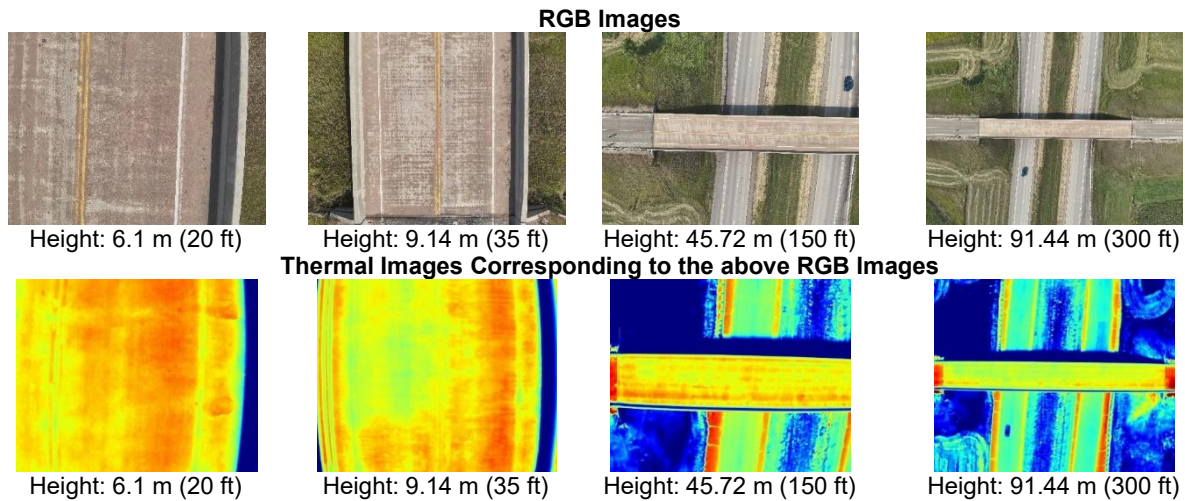


Figure ES.19 – Sample of Drone RGB-Thermal Images Collected at Different Heights from Bridge Decks

ES.11.2.1 Gimble Angle

To better stitch individual images after collecting multiple images of a bridge deck, it is important to keep the gimble pointed directly downward (Nadir) instead of having a slanting angle (**Fig. ES.20**). **Figure ES.21** provides sample images captured with different gimble angles. The images collected at a slanting angle cause problems in image stitching procedure. For example, **Fig. ES.22** compares stitched image of a bridge based on the data collected at a slanting angle with a stitched image obtained from images collected at the recommended angle.

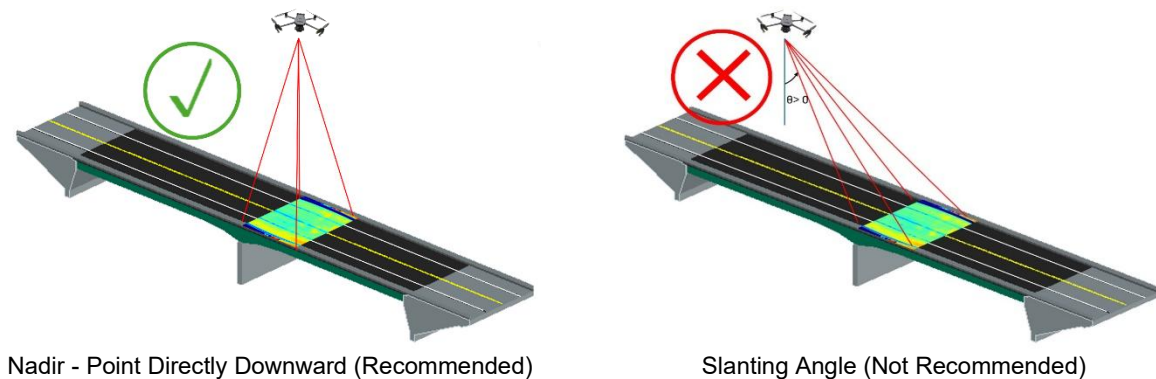


Figure ES.20 – Recommended Gimble Angle for RGB and Thermal Imaging

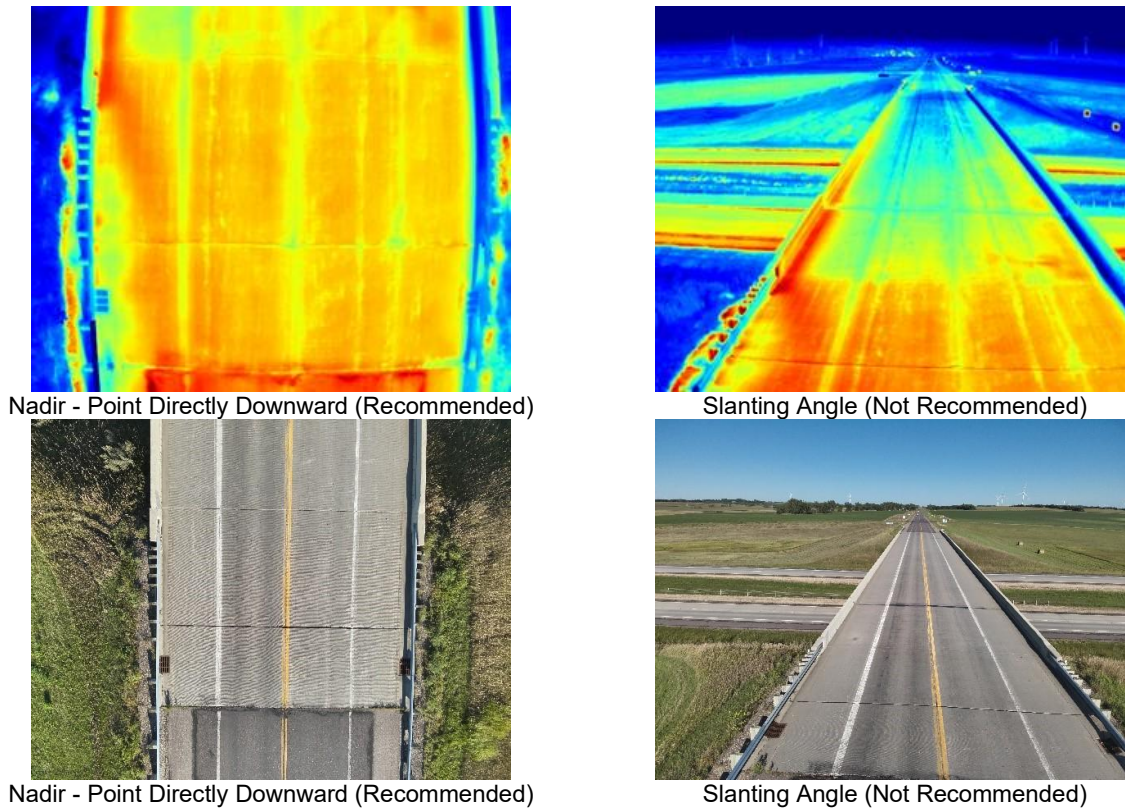


Figure ES.21 – Sample Images with Different Gimble Angles

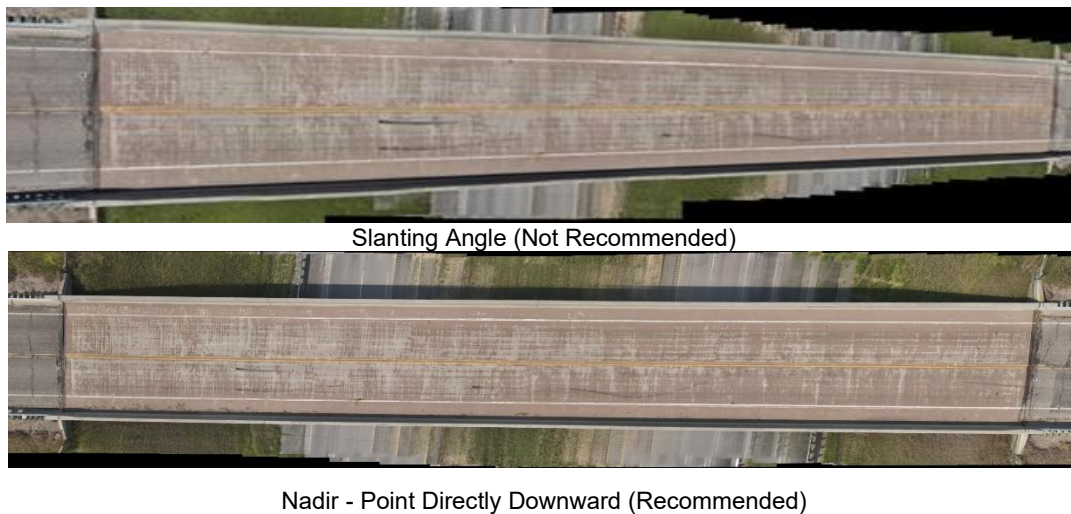


Figure ES.22 – Comparison of Stitched Images Based on Data Collected at Slanting Angle and Pointing Downward

ES.11.2.2 Overlap Between RGB or Thermal Images

There should be an overlap between the successive RGB or thermal images of bridge decks. An overlap of 30 to 60% is recommended (**Fig ES.23**). Note that since the field of view of existing thermal cameras is narrower than that of RGB, more images might be needed when collecting thermal data. Adequate overlap provides a better image stitching outcome for bridge decks.

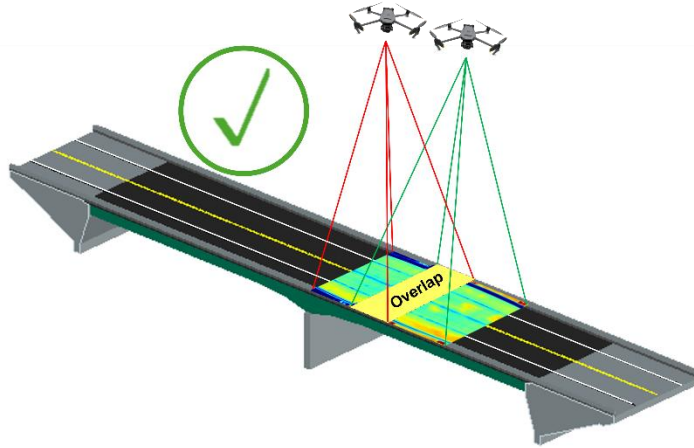


Figure ES.23 – Required Overlap in Successive Drone Imaging

ES.11.2.3 Suitable Time Period to Collect Thermal Images of Bridge Decks

Bridge inspections usually start early in the morning and last for the whole day. However, thermal gradients of the bridge decks may vary substantially from sunrise to sunset. It is crucial to find the best times of days in which thermal data is viable and sufficient to detect deamination. Note one may collect thermal data at night; however, it is not common, and the neural network trained in the present project works only on the day data, where delaminated areas of concrete are hotter.

To understand the best timeframe for thermal data collection, thermal images were captured for a bridge from 6 a.m. to 8 p.m. with two-hour intervals during a summer day at a latitude of 45.1078764°N (South Dakota, USA). **Figure ES.24** shows the stitched thermal maps of the bridge deck for these 14 hours. At 6 a.m., the delaminated regions were colder as compared to the intact regions of the concrete deck. As the bridge absorbs heat during the daytime, the delaminated regions heat up more than the intact regions thus they appear as hot spots in the images. From noon to 4 p.m., the delaminated regions were conspicuous as easily seen in the figures as hot spots. In the evening, as the bridge got cooler, the delaminated regions started to fade out.

To explore the deep learning model performance on these 14-hour data collection scheme, the central portion of the stitched images, the two traffic lanes, were cropped and then fed to the neural network. **Figure ES.25** shows the model detected delaminated regions. Using the morning data, the neural network was not able to detect correct delaminated regions as the temperature of these regions was lower than the intact regions. However, the model was consistent in detecting delamination using data obtained between noon and 4 p.m.

It is understood that location and weather also play a significant role in collecting suitable thermal data for bridges. Nevertheless, as a general guide and based on thermal maps and deep learning model performance to detect delamination, the study proposes to collect thermal images from concrete bridge decks from 11 a.m. to 5 p.m. at latitudes less than 46°N in the northern hemisphere. The time window is expected to be shorter in higher latitudes such as Alaska.

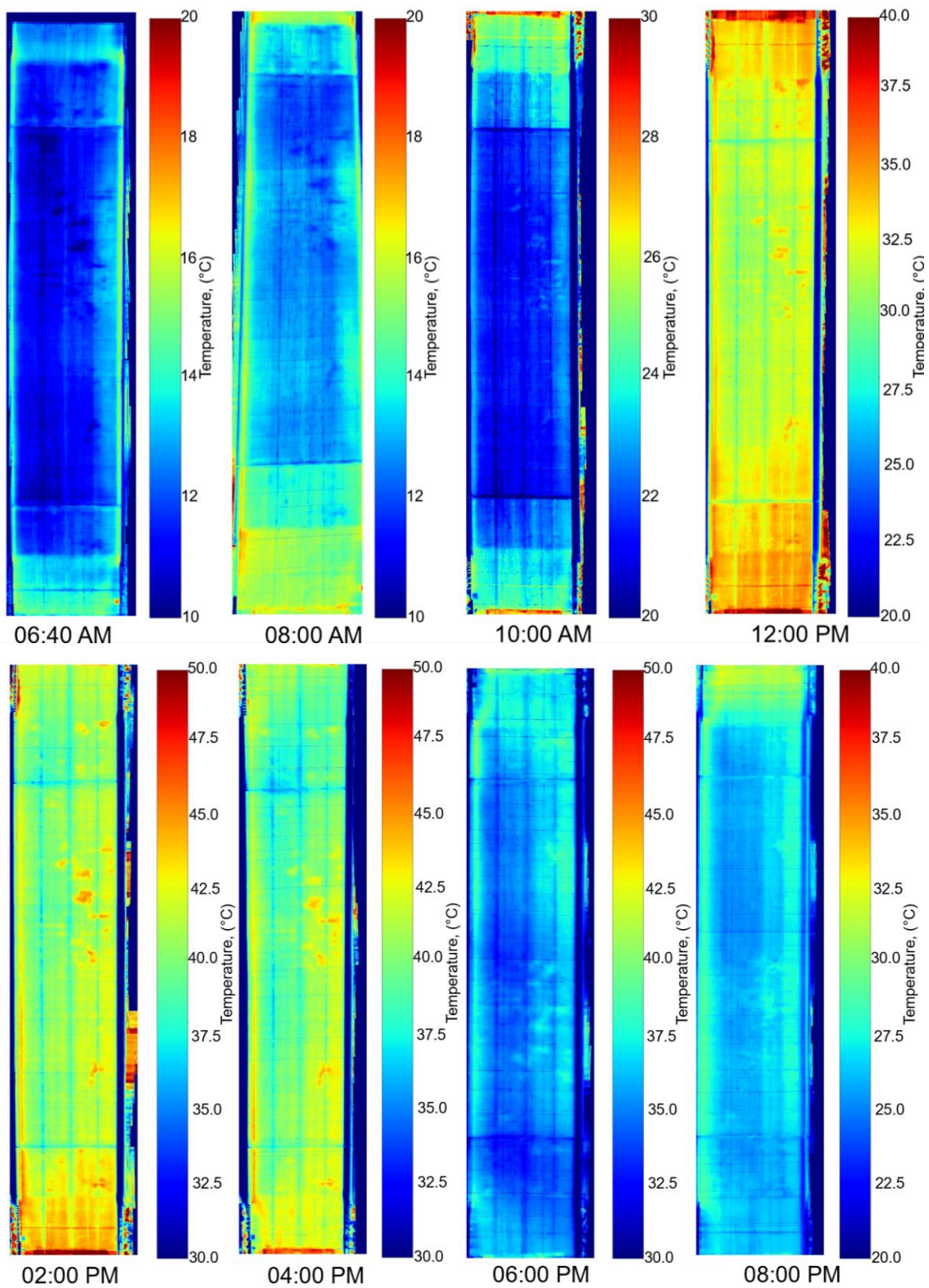


Figure ES.24 – Suitable Time for Thermal Data Collection

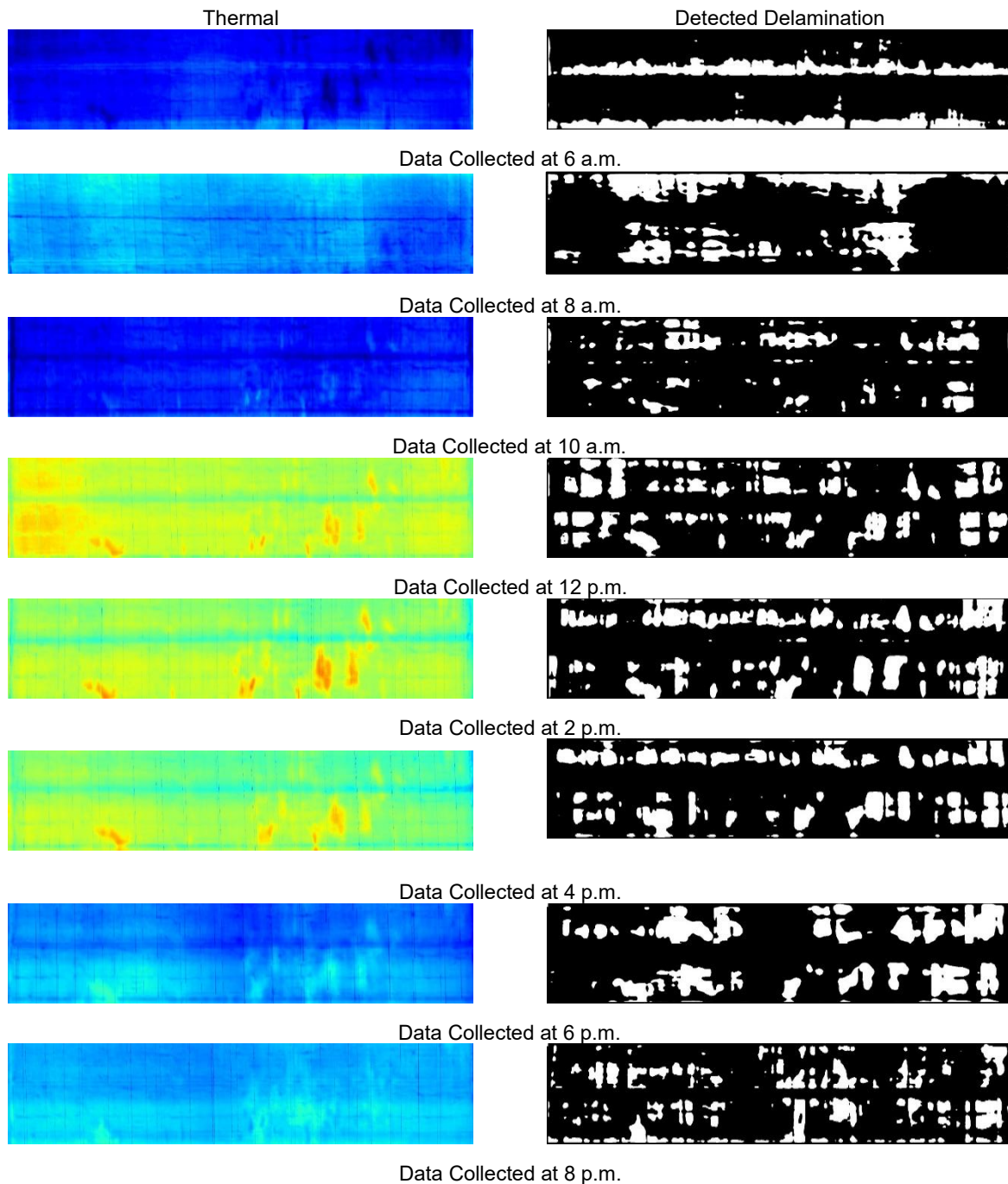


Figure ES.25 – Delamination Detection by Neural Network for Thermal Images Collected at Different Time of a Summer Day

ES.12 Summary and Conclusions

Regular maintenance of transportation infrastructure such as bridges is crucial in extending their service life while reducing long term repair costs. Transportation agencies such as the Federal Highway Administration (FHWA) seek emerging technologies and innovative programs to enhance asset management and maintenance. Furthermore, timely inspections of bridges by trained personnel are mandated per national and state specifications. The Alaska Department of Transportation and Public Facilities (DOT&PF) engineers inspect approximately 500 bridges each year. In Alaska, approximately 44% of bridges are in good condition, 49% are in fair condition, and 7% are rated poor based on the 2023 National Bridge Inventory (NBI) data.

In general, visual inspection is the common practice for inventory and routine inspections and is combined with other tools such as nondestructive evaluation (NDE) in other inspections for enhanced assessment. The inspector must complete both an NBI inspection and an element level inspection for each bridge. Nationwide, most steps of bridge inspections such as data collection, defect quantification, and damage reporting are done manually. Furthermore, each of these steps is time consuming, error prone, and hard to repeat in the following inspections due to the manual nature of these activities. Such measurements often require traffic control for the safety of the inspection crew, and the findings might not be consistent if done by different inspectors.

Computer vision, a field of artificial intelligence (AI) that can analyze scenes at human level performance, can speed up defect identification and quantification only using images of bridge elements. Progress in computer aided analysis of visual imagery has initiated widespread use of computer vision tools in various tasks, including but not limited to object classification, categorization, detection, and generation by learning from provided datasets for specific tasks. Even though the academic and industry level developments are interesting and on the right track for field deployment, they usually lack practicality and are overwhelming for everyday users. For example, the academic-level computer programs are not as user-friendly as commercial software, and the industry-level software packages are so enlarged targeting multiple assets preventing them from being practical. In summary, current tools do not fully address the needs of DOTs for routine bridge inspections.

The primary objective of this project was to develop practical AI tools that help inspectors with defect detection and measurement and facilitate the inspection reporting following standard practices. To achieve the project goals, a comprehensive literature review of the current scientific trend and AI tools was performed on delamination detection, bridge map generation, and damage state estimation. Then, a new definition of condition states for delamination of concrete bridge decks and steel girder defects suitable for computer programming was proposed. Subsequently, more than 50 bridges were inspected, and a comprehensive dataset of inspector annotated delamination was compiled using iPhone-based lidar, conventional cameras, and drone based RGB and thermal cameras. Images of steel girders with and without damages were also collected. A novel approach to generate a thermal map for entire bridge using RGB and thermal images from drones was developed. Calibration of RGB and thermal cameras was carried out to undistort the images. Homography matrices obtained from RGB image stitching were utilized to generate thermal maps of the entire bridge decks. Best practices to collect lidar, RGB and thermal data from concrete bridge decks were also proposed. Finally, an AI-based tool that can detect, quantify, and map the inspector's annotations, and produce a delamination map of whole

bridge deck including their condition states was developed for RGB images. A separate AI-based tool was developed for thermal images to identify the potential delaminated regions. Since data normalization usually impacts training and model accuracy, seven different normalization methods based on the mean, standard deviation, minimum and maximum temperatures for thermal images were evaluated to obtain the best prediction of delamination. Furthermore, for inspecting bridge steel girders, a simple tool with a human-in-the-loop method was developed to allow users to remove camera distortions from RGB images, annotate steel girder defects such as paint loss in the image, and obtain the physical area of the damage. A user-friendly software, Bridge Computer Vision Inspection Tools (Br.CVIT), was developed by the research team implementing all of the above-mentioned methods and approaches.

The following conclusions can be drawn from this computer vision-based study on bridge inspections:

- A new condition state definition was developed for delamination of concrete bridge decks. The new definition enhances the physical interpretation of existing criteria in AASHTO by using definitive and quantitative measures at all four levels that are suitable for computer programming and automated analysis. The new definition ensures greater alignment between manual assessments and computer-based analyses, improving the accuracy and consistency of damage evaluations. The definitions were implemented in Br.CVIT as default values and was found viable and repetitive.
- A similar definition was developed for steel girders and were implemented in Br.CVIT.
- A new image-lidar dataset of inspector annotated delamination was compiled after inspecting more than 50 bridges. The images are particularly applicable in training neural network models to automatically detect and measure inspector-annotated delaminated areas. This dataset contains 1,918 RGB photographs of inspector annotations and 18 lidar scans of bridge decks.
- For detections of the inspector annotations on bridge decks in RGB images, three neural network models for image segmentation (U-Net, Attention U-Net, and Recurrent Residual U-Net) were evaluated through hyper-parameter tuning, image preprocessing, and augmentation. The color and geometric augmentations of a somewhat small training dataset expanded the dataset to 70,000 to improve generalization and accuracy. U-Net was found to be the best model.
- To detect delamination in thermal images, six neural networks for image segmentation (U-Net, Attention U-Net, Residual U-Net, Transformer ResU-Net, Recurrent Residual U-Net, and ResUNet++) were evaluated. U-Net was found to be the best model.
- Among seven methods to normalize the thermal image dataset, the normalization method based on the individual mean temperature and standard deviation performed better.
- A method to generate high-quality bridge deck maps was developed to extract 2D bridge deck maps from reconstructed 3D mesh (point cloud) data. This allows high accuracy in quantification of the annotated area by inspectors.
- Thermal maps for several bridges were successfully generated using a novel approach based on RGB and thermal image stitching.
- A two-level assessment of the detection of delaminated areas in bridge decks using lidar data was conducted. Local delamination assessment, which evaluates pixel-to-pixel correspondences, was conducted to measure the quality of the granular prediction. The pixel-to-pixel level accuracy for RGB image was less than 75% for the best model. The

global delamination assessment (area-to-area) exhibited, on the other hand, a near-to-perfect performance with 99.5% accuracy over 18 bridges. Furthermore, the global assessment confirmed that, regardless of the quality of the local assessment, the framework can quantify the area inside the delaminated spot with high accuracy. The accuracy of the neural network in detecting delaminated areas of 2D RGB maps of bridge decks obtained from drone RGB images was at 96%.

- For thermal image dataset, the best U-Net model showed an accuracy of 77% to 83% in terms of intersection over union to predict delamination in concrete bridge decks. As a global performance assessment, it was found that the total physical delaminated area of bridge decks detected by the AI model was on average 96% that of the ground truth for seven concrete bridge decks, indicating that the model is very accurate.
- The proposed defect detection method for steel girders was also simple and reasonably accurate.

Overall, the proposed multi-device computer-vision based study can expediate bridge deck inspections for delamination by automating the entire process of data analysis for point clouds, RGB, and thermal images, and damage identification and condition state estimation using neural networks.

CHAPTER 1 – INTRODUCTION

1.1 Introduction

There are more than 617,000 bridges in the United States. Of which, 42% have already passed their design life (ASCE, 2021). Furthermore, 44.10% of the US bridges are in good condition, 49.14% are in fair condition, and 6.75% are in poor condition based on the 2024 National Bridge Inventory (NBI) data (Infobridge, 2024). Bridges in the USA are required to be routinely inspected every two years (NBIS, 2022). The nation-wide bridge inspection data is used to allocate federal funds for maintenance and repair needs in the following year.

The Alaska Department of Transportation & Public Facilities (DOT&PF) is responsible for condition assessment of approximately 1000 bridges in the state. Of which, approximately 44% are in good condition, 49% are in fair condition, and 7% are rated poor, well aligned with the national trends (Infobridge, 2024). The Alaska Bridges and Structures Manual (2023) specifies different types of inspections such as inventory, routine, in-depth, and damage. Visual inspection is the common practice for inventory and routine inspections and is combined with other tools such as non-destructive evaluation (NDE) in other inspections for enhanced assessment. The inspector must complete both an NBI inspection (following the FHWA Specifications for the National Bridge Inventory, 2022) and an element level inspection (based on the AASHTO Manual for Bridge Element Inspection, MBEI, 2019) per bridge.

Almost all visible components of a bridge are assessed in a routine inspection by trained personnel. Some bridge elements such as decks are inspected from both top and underneath the bridge and some components such as girders and bents are usually inspected from underneath. Sometimes heavy equipment such as boom trucks is used in a routine inspection to access hard-to-reach areas. Nevertheless, submerged (e.g., bents in water) and embedded (e.g., piles) components are not inspected in a routine inspection. They might be assessed in different inspection types using more sophisticated methods and equipment.

Nationwide, most steps of bridge inspections such as data collection, defect quantification, and damage reporting are done manually. Furthermore, each of these steps is time consuming, error prone, and hard to repeat in the following inspections due to the manual nature of these activities. For example, the “deck delamination mapping” requires manual detection and measurement of delaminated concrete, patched areas, exposed reinforcing steel bars, and spalling. Such measurements often require traffic control for the safety of the inspection crew, and the findings might not be consistent if done by different inspectors. In recent years, drones equipped with various sensors (e.g., Red-Green-Blue (RGB), thermal, and/or lidar) have been specifically developed for bridge inspections with costs that are affordable by many transportation agencies (Feroz & Abu Dabous, 2021). When a drone with RGB and thermal cameras is used in a routine inspection, the bridge data can be quickly collected without the need for heavy equipment and with a minimal impact to traffic. However, data (image) processing obtained by drones to detect and quantify bridge defects are also carried out mostly manually or using proprietary software.

Overall, bridge evaluation through routine inspection is performed with conventional equipment (e.g., inspection trucks, cameras, laser distance meters, and drones in recent years) and the quantification of observed defects (e.g., area of spalled concrete) are mostly done manually.

Of different bridge components, concrete decks and steel girders are the focus of this study. For decks, inspectors usually look for major cracks, spalling, patching, and delamination. The deck surface damage is visible and is usually documented using cameras. However, detection of subsurface deck defects such as delamination is more challenging. Different NDE techniques have been utilized for deck subsurface damage detection including chain drag, impact echo, ground penetrating radar (GPR), electric resistivity, and infrared thermography (Aljagoub et al., 2022; and Omar & Moselhi, 2022). The chain-drag method, which is the simplest and most used method, is done by dragging a few strings of steel chains on the deck surface to hear sounds that are different when reaching delaminated areas. Nonetheless, the chain-drag method is subjective in nature and depends on the expertise of the inspector and ambient noise (Aljagoub et al., 2022). Collecting data using impact echo, GPR, electric resistivity, and infrared thermography requires special equipment and is usually laborious, and post-processing of such data requires expertise and proprietary software.

The same issues discussed above exist for bridge steel girders to detect and quantify their defects. As mentioned earlier, even when drones are used to collect the image of girders, processing such data (e.g., image) to obtain and quantify girder defects is mostly manual and subjective.

Computer vision, a field of artificial intelligence (AI) that can analyze scenes at human level performance, can speed up defect identification and quantification only using images of bridge elements. Furthermore, this and other AI tools can be utilized to expedite and unify reporting. Even though the academic and industry level developments are interesting and on the right track for field deployment, they usually lack practicality and are overwhelming for everyday users. For example, the academic-level computer programs are not as user-friendly as commercial software, and the industry-level software packages are so enlarged targeting multiple assets preventing them from being practical.

In summary, current tools do not fully address the needs of DOTs for routine bridge inspections. Alternative data collection, data processing, defect detection and quantification, and digital reporting are needed to expedite and enhance bridge inspections.

1.2 Problem Statement and Research Objective

As discussed above, despite novel inspection procedures and emerging technologies, bridge inspections in Alaska and nationwide remain predominantly manual. For example, defect detection and quantification such as deck delamination mapping are done manually, which require significant time and labor. Furthermore, existing NDE methods are not common in routine inspection since they are also time consuming, costly, and require expert input for data processing. AI-based inspection tools are emerging, but the current tools are either too complex for routine use or not tailored to DOT needs. The main objective of the present project was to develop practical AI tools that help inspectors with defect detections and quantifications, and to facilitate the inspection reporting following the DOT&PF standard requirements. Concrete decks and steel girders were the focus of the study.

1.3 Scope of Study and Document Outline

To achieve the project goals, more than 50 bridges were inspected by the research team using multiple devices (Light Detection and Ranging (lidar) sensors of iPhone/iPad, hand-held cameras, and drones with RGB and thermal sensors) and data format (point clouds, RGB, and thermal images), additional data was provided by DOT&PF, a large inspection database including photographs of the selected elements with and without damage was compiled, and a user-friendly software powered by computer vision tools was developed for the selected elements to detect their defects, quantify the defects, and produce a report following standard practices. The present report includes a summary of these activities.

Chapter 2 presents a comprehensive literature review of the current scientific trend and AI tools on delamination detection, bridge map generation, and damage state estimation. Then, new definitions of condition states for delamination of concrete bridge decks and steel girder defects suitable for computer programming are proposed in **Ch. 3**. **Chapter 4** discusses the comprehensive inspection dataset of inspector annotated delamination and steel girder damages. The development of an AI-based inspection software (Br.CVIT) that can detect, quantify, and map the inspector's annotations, and produce a delamination map of whole bridge deck including their condition states is discussed in **Chapters 5-8**. **Chapter 7** provides a human-in-loop method of damage quantification for steel girders, which was also included in the software. Finally, the best practices to collect lidar, RGB and thermal images of the bridge deck are discussed in **Ch. 9**. Finally, **Ch. 10** provides a summary and conclusions of the study.

CHAPTER 2 – LITERATURE REVIEW

2.1 Introduction

Routine inspections are necessary to monitor condition states of bridges. Both a National Bridge Inventory (NBI) inspection (a 10-level condition rating for bridges and bridge components (e.g., decks, superstructures, substructures, etc.) following the FHWA Specifications for the National Bridge Inventory, 2022) and an element-level inspection (a four-level element-by-element condition state assessment based on the AASHTO MBEI, 2019) must be carried out per bridge. Based on the inspection findings, bridges are prioritized, and funds are allocated to better maintain or repair them. In the United States, routine inspections of bridges are performed every two years resulting in approximately 300,000 inspections per year. Inspection seasons are usually in the warmer months of the year such as May through October but vary state by state.

In current practice, bridges are inspected mostly manually, and their defects are also quantified manually. Visible damage is easy to detect; however, internal defects such as concrete delamination and steel bar corrosion are hard to find. Delamination is an internal defect in concrete caused by insufficient pouring, fatigue, freeze-thaw cycles, and environmental influences. Delamination can accelerate the corrosion of reinforcing steel bars, which further causes internal pressure then cracking at the concrete-to-bar interfaces. Two simple and cost-effective nondestructive evaluation (NDE) techniques that are widely used in routine bridge inspections to detect concrete delamination are chain-drag and sound-hammering methods. Other NDE methods exist but they are not common in routine inspections.

Drones are getting more attention and are used more frequently nowadays in bridge inspections, especially that drones equipped with both Red-Green-Blue (RGB) and thermal cameras are currently commercially available with a cost of only a few thousand dollars, which is insignificant compared with other NDE methods and/or inspection equipment. RGB data reveals visible damage of bridge elements such as decks, and thermal data can be used to detect subsurface concrete anomalies such as old patching and delamination.

In this chapter, a brief review of element-level inspection and assessment is presented first, then NDE methods for bridge decks are discussed, and a short review of current trends in computer vision, a field of artificial intelligence (AI), specific to bridge inspections is presented at the end.

2.2 Bridge Element Inspection and Assessment

The AASHTO Manual for Bridge Element Inspection (AASHTO MBEI, 2019) provides four-level condition state assessments for all elements of bridges. Since a few bridge elements (concrete decks and steel girders) are the focus of the present study, the AASHTO MBEI element-level assessments are adopted herein over the NBI component-level condition rating. The following sections present the condition levels for a few select bridge elements.

2.2.1 Bridge Decks

The AASHTO bridge deck inspection guideline mostly focuses on the visible defects of deck elements such as spalling, patching, and exposed bars. Deck delamination is the only subsurface defect, which is found using one of the NDE methods. **Table 2.1** presents the AASHTO

definition of condition states for concrete decks.

Table 2.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination /Spall /Patched Area (1080)	None.	Delaminated. Spall 1 in. or less deep or 6 in. or less in diameter. Patched area that is sound.	Spall greater than 1 in. deep or greater than 6 in. diameter. Patched area that is unsound or showing distress. Does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Exposed Rebar (1090)	None.	Present without measurable section loss.	Present with measurable section loss but does not warrant structural review.	
Efflorescence/ Rust Staining (1120)	None.	Surface white without build-up or leaching without rust staining.	Heavy build-up with rust staining.	
Cracking (RC) (1130)	Insignificant cracks or moderate width cracks that have been sealed.	Unsealed moderate width cracks or unsealed moderate pattern (map) cracking.	Wide cracks or heavy pattern (map) cracking.	
Abrasion/ Wear (PSC/RC) (1190)	No abrasion or wearing.	Abrasion or wearing has exposed coarse aggregate but the aggregate remains secure in the concrete.	Coarse aggregate is loose or has popped out of the concrete matrix due to abrasion or wear.	

2.2.2 Superstructure Elements

Superstructure elements such as girders are considered part of the main load-bearing components of a bridge and are inspected for cracks, corrosion, and loss of protective coating. **Table 2.2** presents the AASHTO condition states for prestressed concrete girders and **Table 2.3** presents such definitions for steel girders.

Table 2.2 – AASHTO Condition States for Prestressed Concrete Girders (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination/Spalls/ Patch Areas (1080)	None.	Delaminated. Spall 1 in. or less deep or 6 in. or less in diameter. Patched area that is sound.	Spall greater than 1 in. deep or greater than 6 in. diameter. Patched area that is unsound or showing distress. Does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Exposed Rebar (1090)	None.	Present without measurable section loss.	Present with measurable section loss but does not warrant structural review.	
Exposed Prestressing (1100)	None.	Present without section loss.	Present with section loss but does not warrant structural review.	
Efflorescence/Rust Staining (1120)	None.	Surface white without build-up or leaching without rust staining.	Heavy build-up with rust staining.	
Cracking (PSC) (1110)	Insignificant cracks or moderate-width cracks that have been sealed.	Unsealed moderate width cracks or unsealed moderate pattern (map) cracking.	Wide cracks or heavy pattern (map) cracking.	
Abrasion/Wear (PSC/RC) (1190)	No abrasion or wearing.	Abrasion or wearing has exposed coarse aggregate but the aggregate remains secure in the concrete.	Coarse aggregate is loose or has popped out of the concrete matrix due to abrasion or wear.	
Distortion (1900)	None.	Distortion does not require mitigation or mitigated distortion.	Distortion that requires mitigation that has not been addressed but does not warrant structural review.	
Settlement (4000)	None.	Exists within tolerable limits or no observed structural distress.	Exceeds tolerable limits but does not warrant structural review.	The element has impact damage. The specific damage caused by the impact has been captured in CS 4 under the appropriate material defect entry.
Scour (6000)	None.	Exists within tolerable limits or has been arrested with effective countermeasures.	Exceeds tolerable limits but is less than the critical limits determined by scour evaluation and does not warrant structural review.	
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in CS 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in CS 3 under the appropriate material defect entry.	

Table 2.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Corrosion (1000)	None.	Freckled rust. Corrosion of the steel has initiated.	Section loss is evident, or pack rust is present but does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Cracking (1010)	None.	Crack that has self-arrested or has been arrested with effective arrest holes, doubling plates, or similar.	Identified crack that is not arrested but does not warrant structural review.	
Connection (1020)	Connection is in place and functioning as intended.	Loose fasteners or pack rust without distortion is present but the connection is in place and functioning as intended.	Missing bolts, rivets, or fasteners; broken welds; or pack rust with distortion but does not warrant a structural review.	
Distortion (1900)	None.	Distortion not requiring mitigation or mitigated distortion.	Distortion that requires mitigation that has not been addressed but does not warrant structural review.	
Settlement (4000)	None.	Exists within tolerable limits or arrested with no observed structural distress.	Exceeds tolerable limits but does not warrant structural review.	
Scour (6000)	None.	Exists within tolerable limits or has been arrested with effective countermeasures.	Exceeds tolerable limits but is less than the critical limits determined by scour evaluation and does not warrant structural review.	The element has impact damage. The specific damage caused by the impact has been captured in CS 4 under the appropriate material defect entry.
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in CS 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in CS 3 under the appropriate material defect entry.	

2.2.3 Substructure Elements

Substructure elements include columns, abutments, joints, and bearings. These elements support the structure between spans, connect multiple spans, and allow controlled movement and load transfer between connected components and both ends of the bridge. AASHTO also provides four-level condition states for these elements.

2.3 Bridge Deck Evaluation Techniques

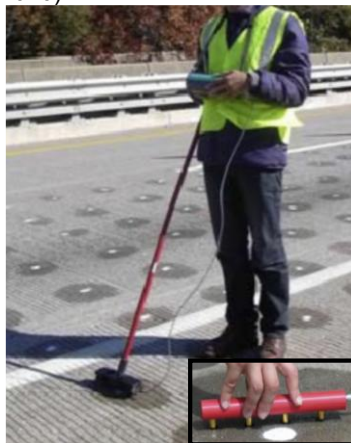
Current bridge inspections heavily rely on manual observation and measurement of visible defects. For subsurface defects, such as concrete delamination or reinforcing steel corrosion, NDE methods are used. For concrete bridge decks, chain-dragging and hammer-sounding are commonly used to distinguish intact from delaminated concrete, but these techniques are labor-intensive and subjective. Other NDE methods such as impact echo, ground penetrating radar (GPR), electrical resistivity, and infrared thermography exist for bridge deck assessment (**Fig. 2.1**), but they are not common in routine inspections.



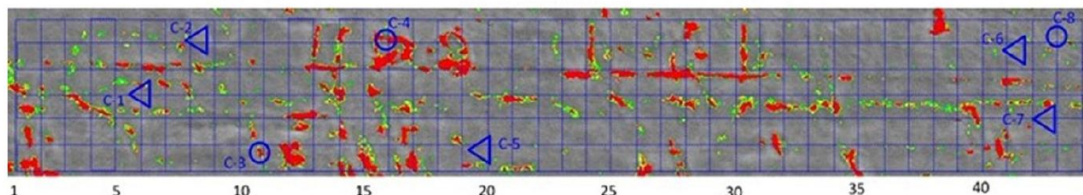
(a) Impact Echo for Delamination Detection in Concrete Decks (Gucunski & Nazarian, 2010)



(b) Use of GPR to Collect Data from Concrete Bridge Deck (Tarussov et al., 2013)



(c) Electric Resistivity for Delamination Detection in Concrete Decks (Gucunski & Nazarian, 2010)



(d) Infrared Thermography (Hiasa et al., 2016)

Figure 2.1 – NDE Methods Used to Inspect Concrete Bridge Decks

The impact echo method (e.g., Gucunski & Nazarian, 2010) detects air pockets, cavities, or delamination in concrete bridge decks by generating acoustic waves in a frequency range of 5-40 kHz. GPR (e.g., Tarussov et al., 2013; Dinh et al., 2016) uses electromagnetic waves to detect anomalies in concrete, such as delamination and corrosion of reinforcing steel bars. The ultrasonic surface wave method (e.g., Gucunski et al., 2012) detects concrete degradation utilizing two receivers, while the half-cell potential methodology identifies corrosion activity in steel reinforcement. Infrared thermography is another NDE method (e.g., Hiasa et al., 2016; Washer et al., 2010) employed to detect delamination in concrete structures by analyzing thermal radiation emitted by objects at temperatures above the absolute zero. Thermal cameras detect the heat emitted by the object and the reflective light from the object's surroundings. Thermography sensors may be either cooled or uncooled, functioning in short to medium wavelengths or long wavelengths.

RGB and thermal sensors can be attached to robots or drones to expedite and automate bridge data collections (**Fig. 2.2**). For example, La et al. (2015) developed a crawling robot that can collect RGB, GPR, electric resistivity, impact echo, and laser data of bridge decks. Charron et al. (2019) used an unmanned ground vehicle equipped with lidar, GPS, RGB and thermal sensors to scan the underside (soffit) of bridge decks. They generated 3D point clouds and colorized them using RGB and thermal images. Furthermore, they created the RGB and thermal maps of bridges with reasonable accuracy.

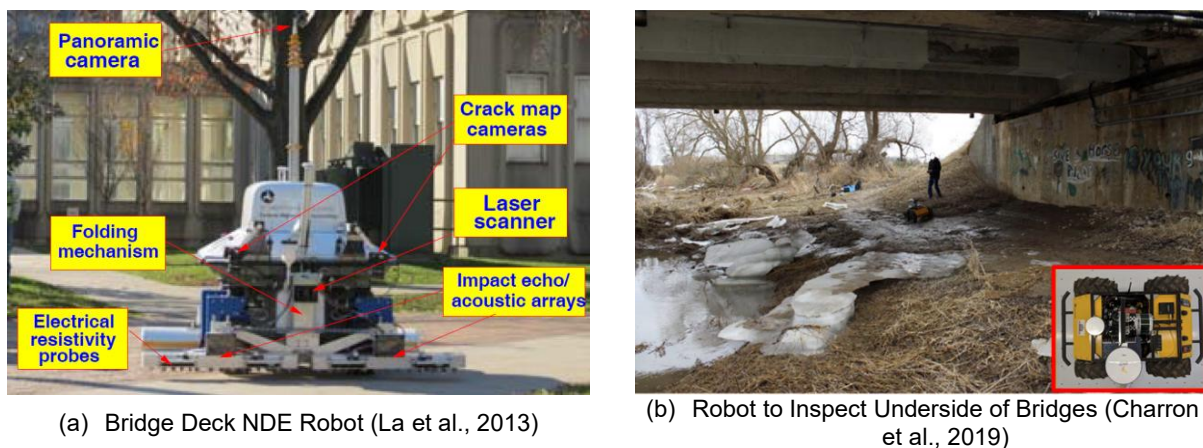


Figure 2.2 – Use of Robots to Inspect Bridges

For thermal image processing, more than ten computational methods have been developed in the literature to detect concrete delamination such as thermal contrast, thermal gradient, adaptive threshold, level-set method, and AI. **Figure 2.3** shows analyses of a thermal image of a bridge deck using a few of these methods. The thermal contrast method has been widely used in past studies (e.g., Washer et al., 2010; Khan & Bartoli, 2015; Tran et al., 2018; Mac et al., 2019; Vyas et al., 2019; Robison et al., 2020; Kilic & Caner, 2021; Mac et al., 2021; Raja et al., 2021; Rocha et al., 2021; Zheng et al., 2021; and Pedram et al., 2022) to identify delaminated regions in the thermal images of bridge decks. Cheng and Shen (2020) utilized the level-set methodology for delamination detection in concrete specimens, exhibiting greater accuracy than the k-means clustering method. Grayscale values of thermal images were used to detect delaminated regions. Ichi and Dorafshan (2022) utilized adaptive thresholding and K-Nearest Neighbor (KNN)

clustering techniques to detect delamination in concrete bridge decks, which achieved an accuracy of 69%. They used commercial software to stitch thermal images in grayscale and used thermal intensities (not the actual deck temperature) for delamination analysis.

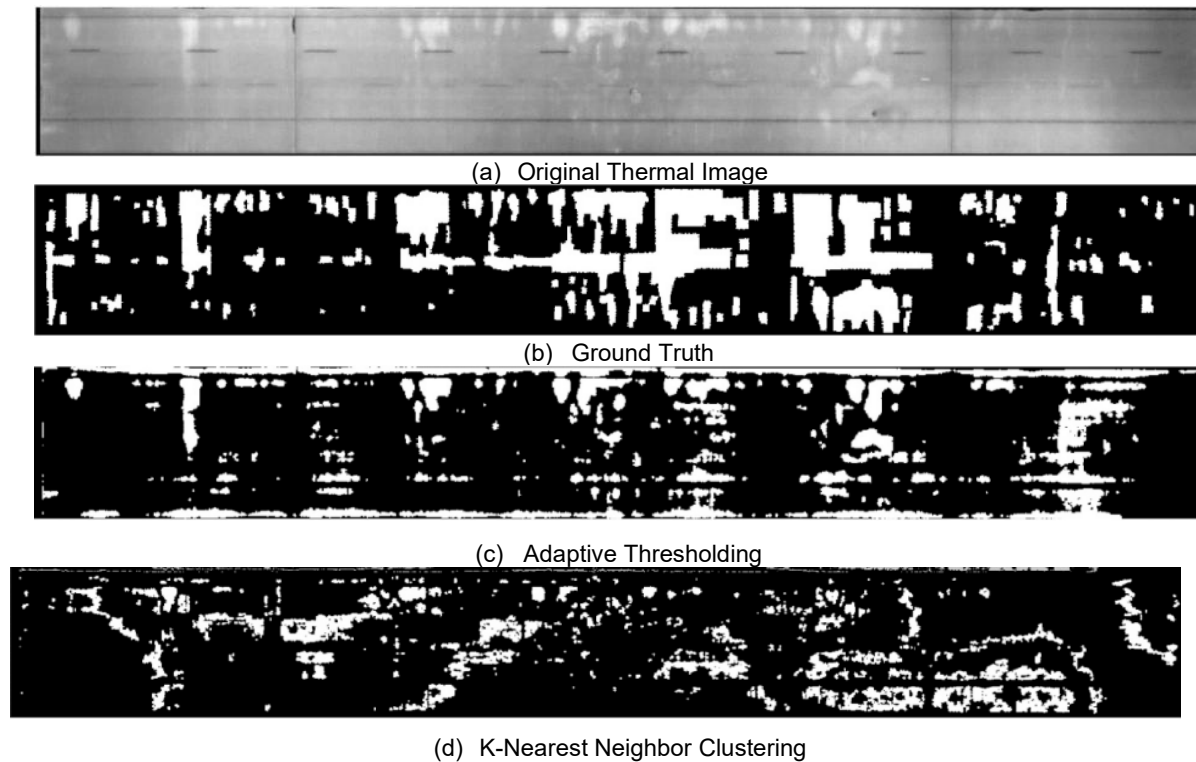
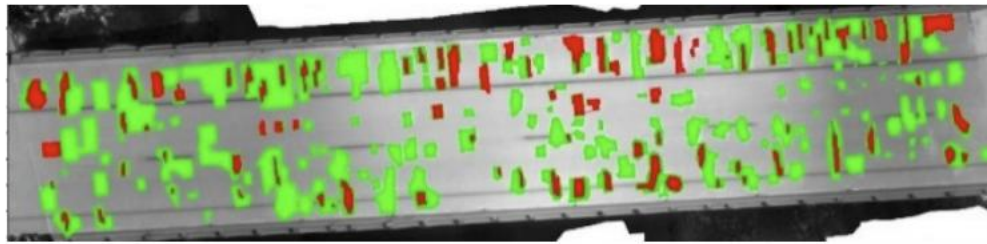
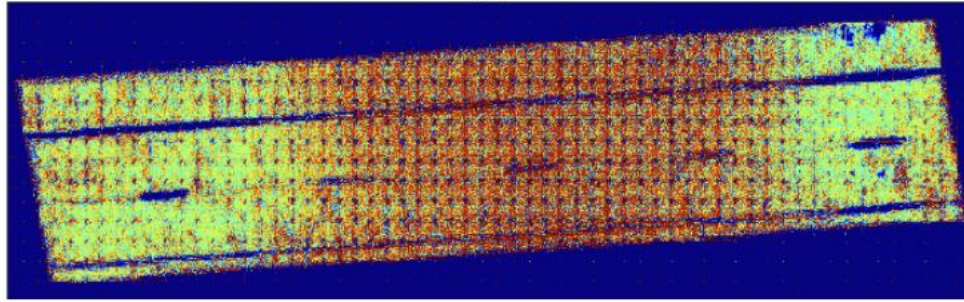


Figure 2.3 – Results of Different Thermal Data Analysis Methods for Bridge Decks (Ichi & Dorafshan, 2022)

A few studies have tried AI techniques to detect concrete delamination in thermal images (**Fig. 2.4**). For example, Cheng et al. (2020) implemented an encoder-decoder neural network using 1,200 thermal images collected from lab specimens (concrete slab with artificial delamination) and reported an accuracy of 83.6% on the lab-based test dataset. They further tested their trained model on actual bridge data using grayscale thermal imaging and achieved an accuracy of 51%. Pozzer et al. (2021) compared the performance of VGG16 (Simonyan & Zisserman, 2014) with MobileNetV2 (Sandler et al., 2018) for delamination detection using 517 thermal images with intensity value (grayscale instead of actual temperature) obtained from bridge sites and reported an intersection over union of 59.5% and 59.1% for VGG16 and MobileNetV2, respectively. Ta et al. (2025) utilized well-established deep learning models (e.g., YOLO, Faster RCNN, SSD) to detect delamination in laboratory specimens. They built 12 concrete specimens (each 56×44×7 in.) with artificial delamination by placing nine polystyrene foams in the concrete slabs and obtained 12,000 thermal images after heating the specimens with halogen lamps, which is not feasible in the field for actual bridges. They also worked with thermal intensity but not the actual temperature. Even though they achieved an accuracy of 96%, the findings of this lab-based study may not be viable in real-world cases due to variations exist in actual bridge conditions and sites. Furthermore, there is a lack of thermal image datasets from actual bridges, as pointed out in SDNET 2021 (Ichi & Dorafshan, 2022), which only includes thermal data of five bridges.



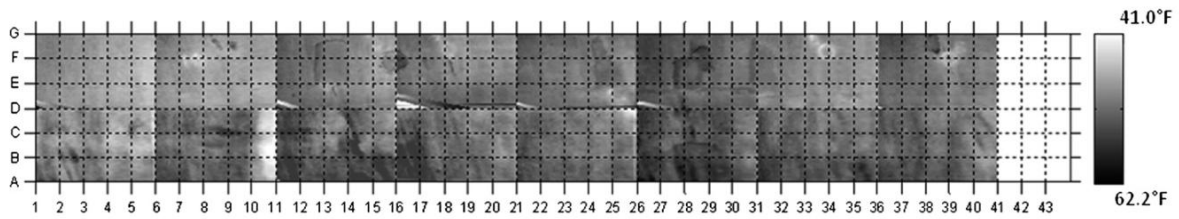
(a) Original Thermal Image (in Gray) with Ground Truth (in Color)



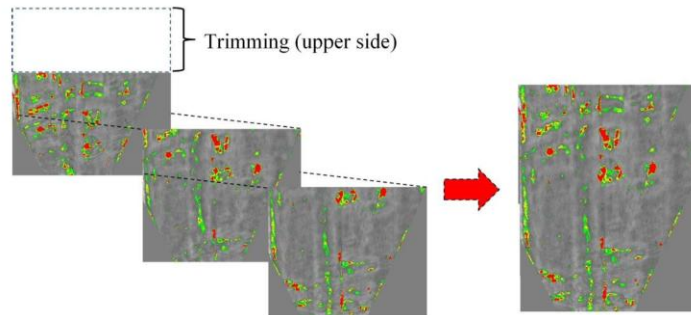
(b) Predicted Delamination Map Using an AI Model

Figure 2.4 – Results of Thermal Data Analysis Using an AI Model (Ichi & Dorafshan, 2024)

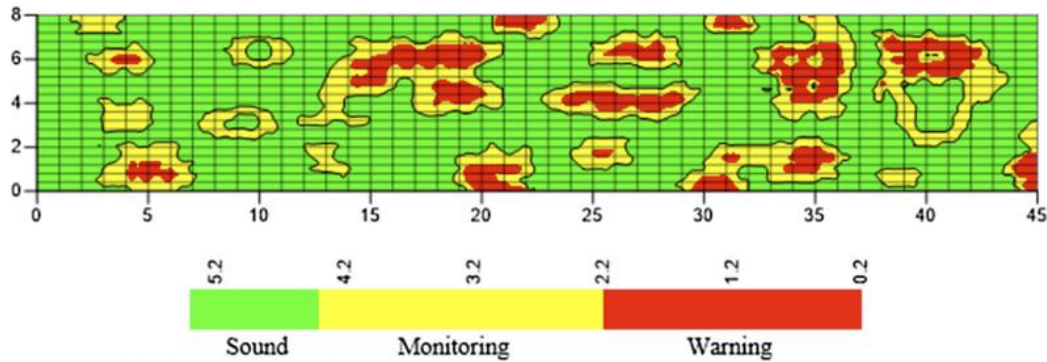
Oh et al. (2012) compared the performance of the infrared thermography with the impact echo method for bridge decks and found a high accuracy for infrared thermography-based inspection compared with the chain-drag method and concrete coring. They manually stitched the thermal images together by placing the images side by side without any stitching algorithm or automation (**Fig. 2.5a**). Hiasa et al. (2016) mounted thermal cameras on a moving car to capture thermal images. They used three different cooled and uncooled thermal cameras mounted on the vehicle and captured the images from specimens placed alongside the road. They also manually trimmed and combined the thermal images together to obtain a thermal map for the entire bridge deck (**Fig. 2.5b**). Omar et al. (2018) generated a thermal map of a bridge deck by manually cropping and then stitching successive thermal images (**Fig. 2.5c**) collected from a camera mounted on a car. They did not remove lens distortions from thermal images. To obtain the delamination map from the thermal map, they employed K-means clustering method to segment thermal images.



(a) Thermal Images Placed Side by Side (Oh et al., 2012)



(b) Thermal Image Stitching Proposed by Hiasa et al. (2016)



(c) Thermal Map Generated Using Manual Method (Omar et al., 2018)

Figure 2.5 – Current Methods of Generating Thermal Maps for Bridge Decks

2.4 Computer Vision for Bridge Element Inspections

With a widespread use of computer vision techniques, the bridge engineering community is exploring its applications in inspection tasks. Matsumoto (2014) used line sensor camera and infrared thermography to detect bridge deck defects and delamination. The framework (**Fig. 2.6**) for bridge inspection and repair consists of three phases. In Phase 1, the deck is scanned using video imagery. Phase 2 involves prioritizing the areas of the deck that are distressed, determining their condition state with a scale of 1 to 4, and planning necessary repairs or rehabilitation. In Phase 3, selected spans of the bridge are scanned from underneath to further assess and address the identified issues.

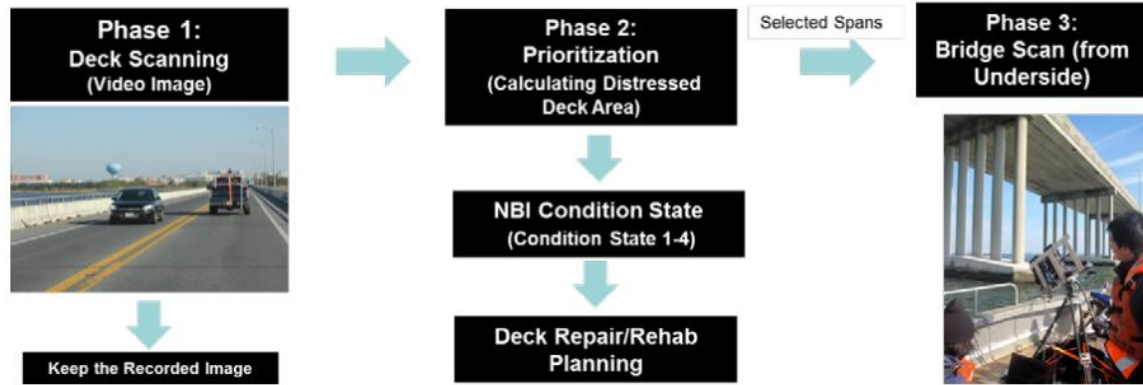


Figure 2.6 – Deck Damage Detection Framework Using Video and Infrared Imaging (Matsumoto et al., 2014)

Won and Sim (2020) detected transverse cracks on bridge decks using optical sensors. They used a three-dimensional (3D) (including height, width, and depth information) computer vision-based localization and mapping technique by utilizing stereovision setup. The proposed framework (**Fig. 2.7**) detects and localizes cracks in bridge decks by employing several innovative techniques. It utilizes multi-view image sequences with a Bag-of-Words representation, eliminating the need for traditional Global Positioning System (GPS) or odometer data for localization. For crack detection, the system uses crack hierarchy and circular histograms instead of conventional methods that rely on line segments and intensities. Additionally, it creates a comprehensive transverse crack dataset based on bespoke ontologies and schemas, covering the entire bridge deck. These advancements aimed at faster deployment and achieving higher accuracy in large-scale crack mapping and measurement. They collected a dataset of 2,638 image patches, consisting of 1,262 images of cracked areas and 1,376 images of uncracked (background) areas of bridges. Two classification approaches were applied to these patches, a conventional method and a crack pixel-based representation approach. A support vector machine (SVM) classifier with a radial basis function kernel was trained using the data, with parameters chosen empirically. Performance, evaluated via 10-fold cross-validation, showed that the conventional method achieved an accuracy of 87.2%, while the crack pixel-based approach achieved a higher accuracy of 95.5%. The results indicate that the crack pixel and segment-based methods offer a more effective description and identification of cracks compared to the conventional approach.

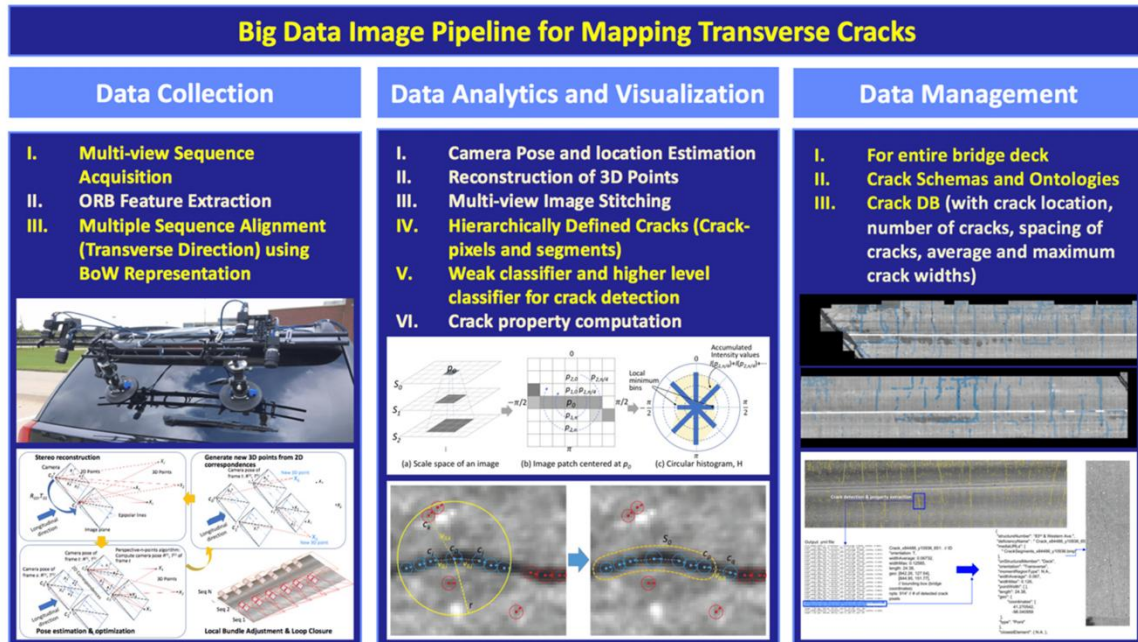


Figure 2.7 – Computer Vision-based Localization and Mapping of Bridge Deck Cracks (Won & Sim, 2020)

Perry et al. (2020) used Structure-from-Motion (SfM) to generate 3D reconstructed model of bridges and to identify target elements. Then they used image processing techniques such as the black hat transform with the Canny edge detection to detect bridge defects. The black hat transform is a morphological operation used in image processing to highlight dark features on a bright background. Specifically, it emphasizes the portions of an image that are smaller than the structuring element used in the transform and darker than their surroundings. The authors also proposed a method to track growth of defects through image comparison (Fig. 2.8).

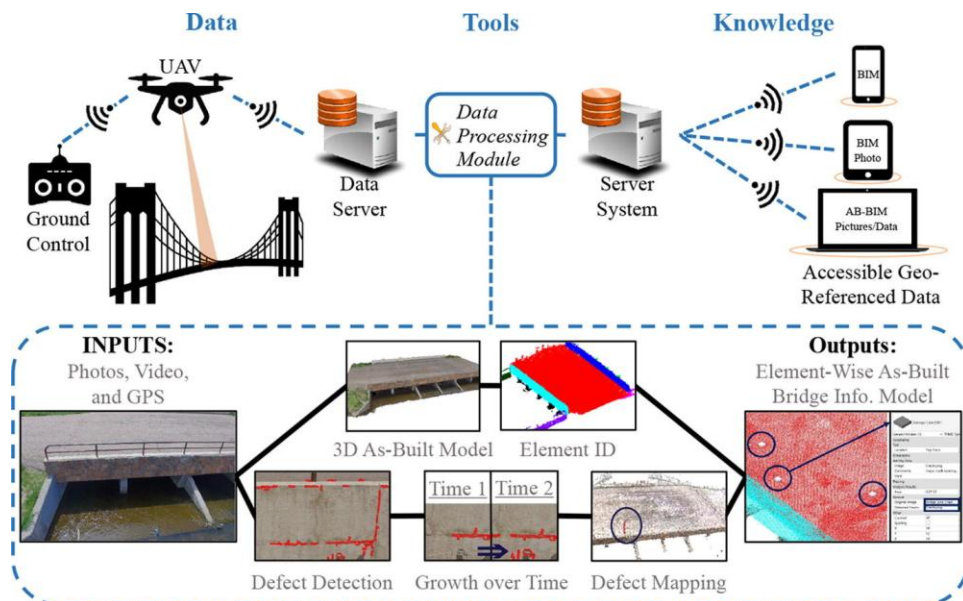


Figure 2.8 – 3D Reconstruction and Defect Detection Framework (Perry et al., 2020)

Ali et al. (2021) used computer vision techniques with a faster region-based convolutional neural network (R-CNN) damage detection framework (**Fig. 2.9**) on data collected using drones for GPS denied locations to detect loosened bolts and steel corrosion. The process begins with identifying the structure to be inspected, followed by deploying an autonomous unmanned aerial vehicle (UAV) equipped with a proportional-integral-derivative (PID) controller for flight stability, an ultrasonic beacon system for precise positioning, and an action camera for capturing visual data. The data was transmitted in real-time using a real-time streaming protocol. A modified Faster R-CNN machine learning model, consisting of a Region Proposal Network (RPN), object proposals, and a Fast R-CNN, was employed for image analysis. The model identifies and highlights structural damage and the ground station processes and analyzes the collected data, completing the inspection process. This integration of robotics, computer vision, and machine learning enables efficient and safe autonomous inspection of structures.

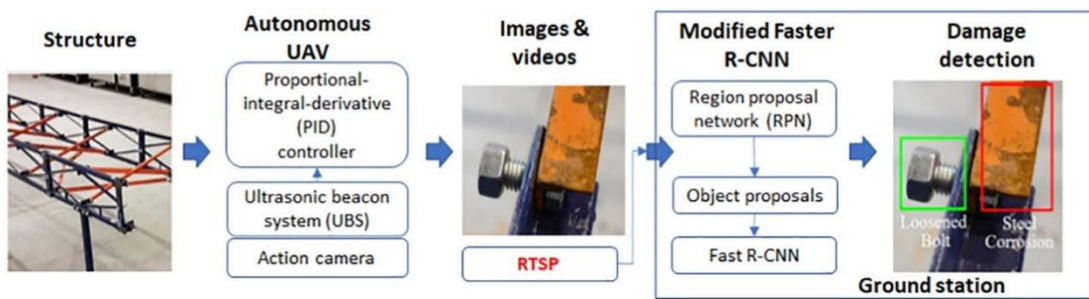


Figure 2.9 – Loosened Bolt and Corrosion Detection Framework (Ali et al., 2021)

Comparing the performance of ZF-Net (Zeiler et al., 2014) and ResNet-101 within the autonomous structural inspection framework, Ali et al. (2021) found that ResNet-101 is more accurate. ZF-Net is an improved version of AlexNet with parameters that were visually examined by reconstructing the learned feature vectors using deconvolution and some other parameters (such as convolutional kernel) were tuned. For steel corrosion detection, ResNet-101 achieved an average precision (AP) of 93.31% and a mean intersection-over-union (IoU) of 90.44%, surpassing ZF-Net's AP of 90.62% and IoU of 88.72%. In detecting steel cracks, ResNet-101 scored an AP of 93.07% and an IoU of 92.24%, whereas ZF-Net obtained an AP and IoU of 85.27%, each. For loosened bolts, ResNet-101 also excelled with an AP of 92.95% and an IoU of 93.79%, compared with the ZF-Net's AP of 88.64% and IoU of 88.64%. The overall mean performance demonstrates that ResNet-101 had a higher mean AP (93.11%) and mean IoU (92.16%) compared with ZF-Net, which had a mean AP of 88.17% and a mean IoU of 87.54%.

Tazarv et al. (2022) used computer vision techniques for inspection and condition assessment of reinforced concrete (RC) bridge columns after earthquakes to automatically determine their conditions. They used deep learning based cascaded detection of deficiencies in their framework using Mask R-CNN. This cascaded detection framework (**Fig. 2.10**) for column damage assessment begins with Stage 1, which detects columns in images. If columns are not detected, the user is prompted to find them manually. Once columns are identified, the process moves to Stage 2 for detecting longitudinal bars, followed by Stage 3 for detecting transverse bars. If both bars are detected, the system reports damage state 4 or 5. Stage 4 involves spalling detection; if spalling is found, it results in damage state 3 or 4. If not, it continues to Stage 5 for crack detection, leading to damage state 1 or 2.

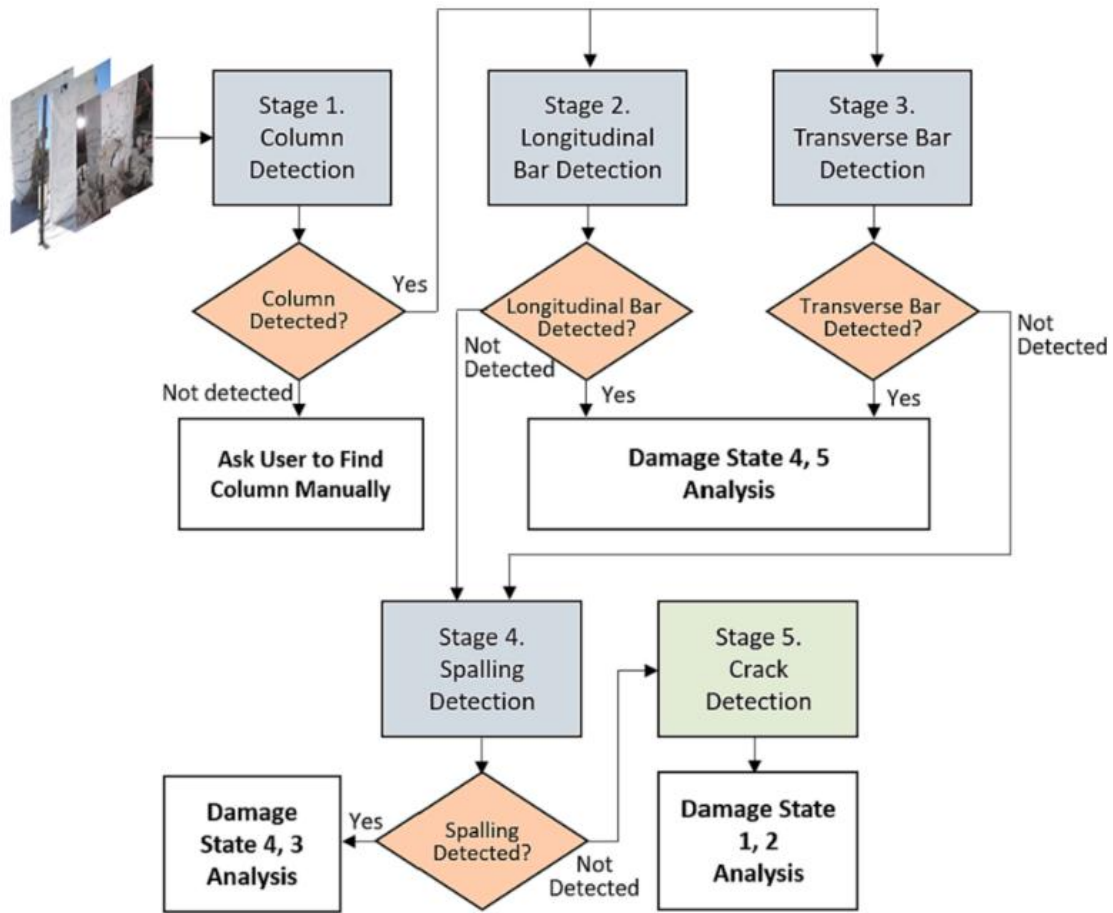


Figure 2.10 – Post-Earthquake RC Column Damage Assessment Framework (Tazarv et al., 2022)

The results of the study by Tazarv et al. (2022) indicate a high performance across all components. For columns, with 41 instances, the localization precision was 90.13% and its recall was 90.91%, while the segmentation precision was 88.90% and the recall was 89.23%. Spalled areas, with 72 instances, showed a localization precision of 95.28% and a recall of 95.88%, with a segmentation precision at 93.97% and a recall at 88.71%. Transverse bars, with 56 instances, had a localization precision and recall of 95.27% and 95.82%, respectively, and a segmentation precision and recall of 92.71% and 93.14%, respectively. Longitudinal bars, with 31 instances, achieved a localization precision of 92.31% and a recall of 92.79%, and a segmentation precision of 91.83% and a recall of 92.17%. The average metrics across all components were 93.24% for the localization precision, 93.85% for the localization recall, 91.10% for the segmentation precision, and 90.81% for the segmentation recall, demonstrating the system's overall effectiveness in the post-earthquake RC column deficiency detection.

Dunphy et al. (2022) developed a multiclass damage detection framework for concrete structures (**Fig. 2.11**) using a combination of Generative Adversarial Networks (GANs) and CNNs. It begins with a multiclass image database of various concrete conditions, such as undamaged, cracked, construction joint, pitting, and spalling. The database was split into labeled and unlabeled datasets. GAN training was performed on unlabeled data, producing a trained

generator and discriminator network. The trained discriminator was then used for transfer learning to initialize the weights and biases of CNN. This CNN was further trained using the labeled data. Finally, hyper-training was applied to the fully connected layers of CNN to optimize it for image classification tasks, leading to accurate identification of different types of concrete damage.

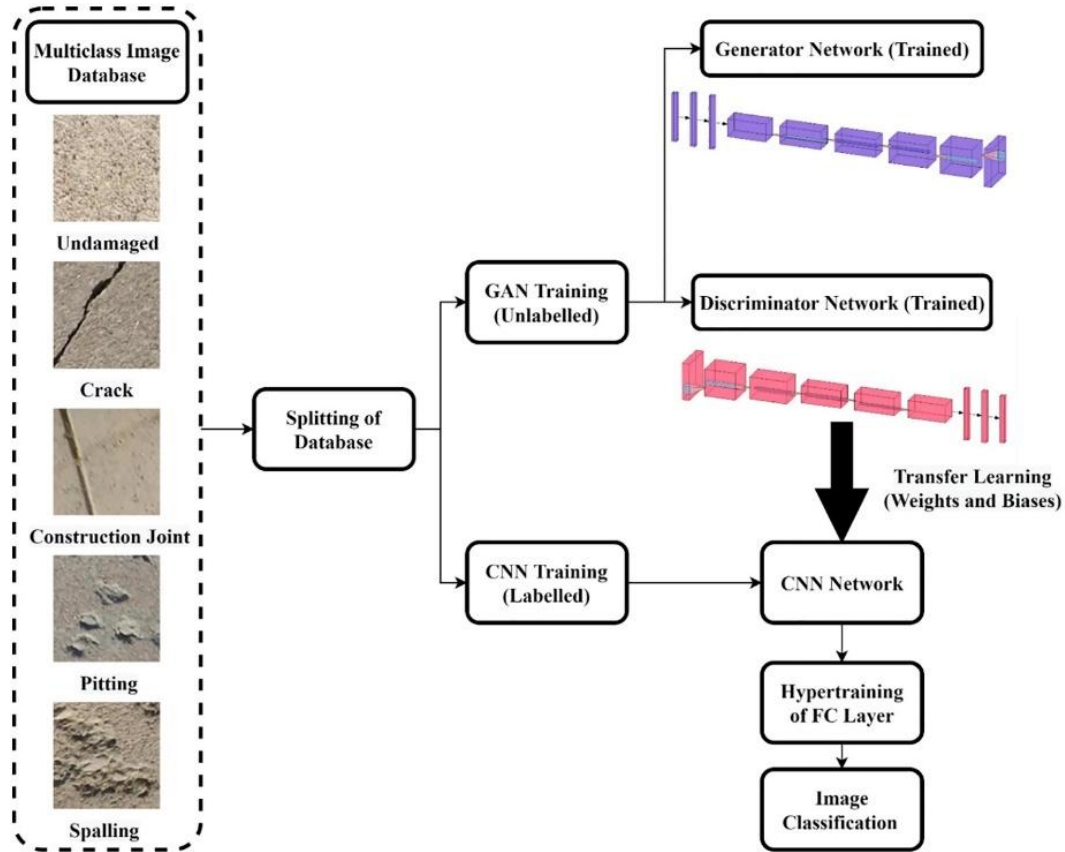


Figure 2.11 – GAN-based Transfer Learning Damage Detection Framework (Dunphy et al., 2022)

Jiang et al. (2023) developed a framework (Fig. 2.12) using object detection models to identify steel bolts and nuts, as well as their conditions, including corrosion and tightness. The proposed method for automatic bridge bolt inspection was a vision-based system leveraging image processing and deep learning algorithms with images captured by UAV. It consisted of three main components: bolt image acquisition, data preprocessing, and bolt damage identification. The image acquisition phase involved using a modified UAV to capture images of bolts from various angles, including the bridge's sides, bottom, and cable clamps. Data preprocessing addressed two key issues: image motion blur due to the UAV's high speed and variations in bolt pixel size caused by differing object distances. To tackle these, an image motion deblurring method using inverse filtering and optical flow was employed to correct motion blur, and an adaptive scale segmentation method with multi-scale template matching and Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) ensured uniform bolt image sizes of 240×240 pixels. Finally, a two-stage bolt damage identification process was used to detect and classify bolt conditions. YOLOv5 network was utilized for bolt detection and segmentation, while EfficientNet classified the bolts into categories such as normal, corroded, or loose. This

method showed high accuracy. For blurred images, it achieved an accuracy of 98.6% with 102 true positives (TP), 36 true negatives (TN), 2 false positives (FP), and no false negatives (FN). For non-blurred images, it attained a 99.6% accuracy with 169 TP, 96 TN, 1 FP, and no FN. The second method, combining preprocessing with a single network, had a lower accuracy of 82.9% for the blurred images (91 TP, 25 TN, 15 FP, 9 FN) and 88.4% for the non-blurred images (150 TP, 86 TN, 19 FP, 11 FN). The third method, using only a single network, had the lowest accuracy, 65.5% for the blurred images (79 TP, 18 TN, 29 FP, 22 FN) and 83.5% for the non-blurred images (143 TP, 79 TN, 27 FP, 17 FN).

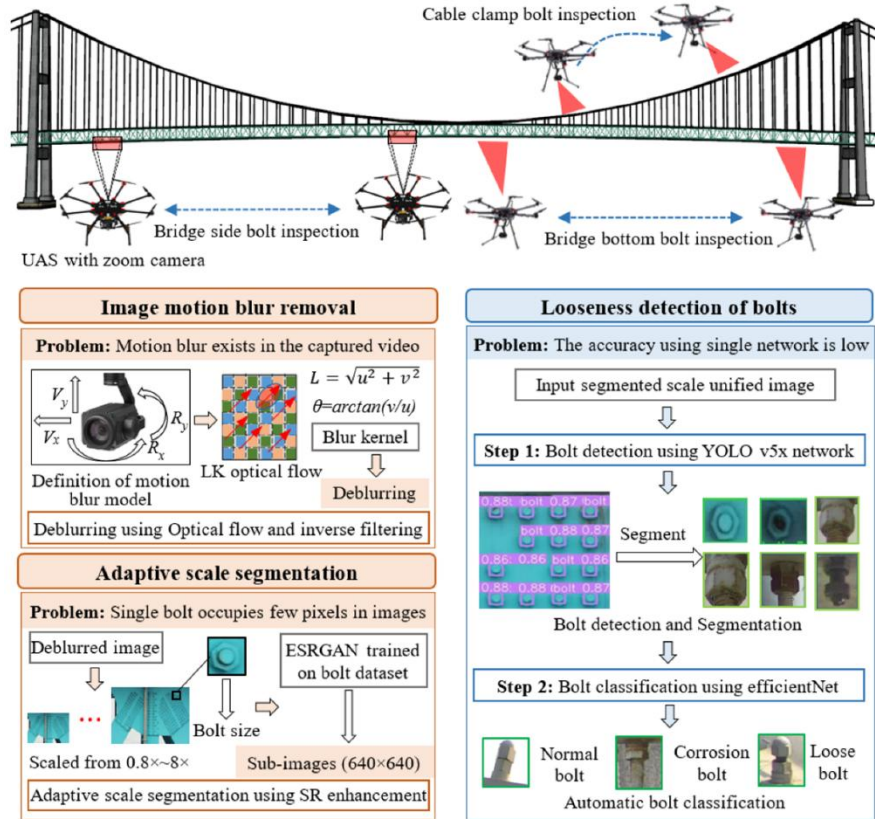


Figure 2.12 – Bolt Detection and Damage Identification Framework (Jiang et al., 2023)

Jiang et al. (2023) proposed a three-stage framework for bridge condition assessment and damage localization using UAV collected data as shown in **Fig. 2.13**. The authors used the YOLOv3 object detection network to detect several types of damage such as cracks, corrosion, and spalling. During data collection, the authors also used visual simultaneous localization and mapping (VSLAM) techniques to localize each damage on bridges by retrieving camera pose with respect to reference world coordinates (**Fig. 2.14**). To achieve this, the authors used stereo-inertial SLAM that used VSLAM with Inertial Measurement Unit (IMU) sensor fusion to increase the robustness of the system. Integration of IMU with VSLAM made the process more robust because of a better estimation of the camera poses for each image.

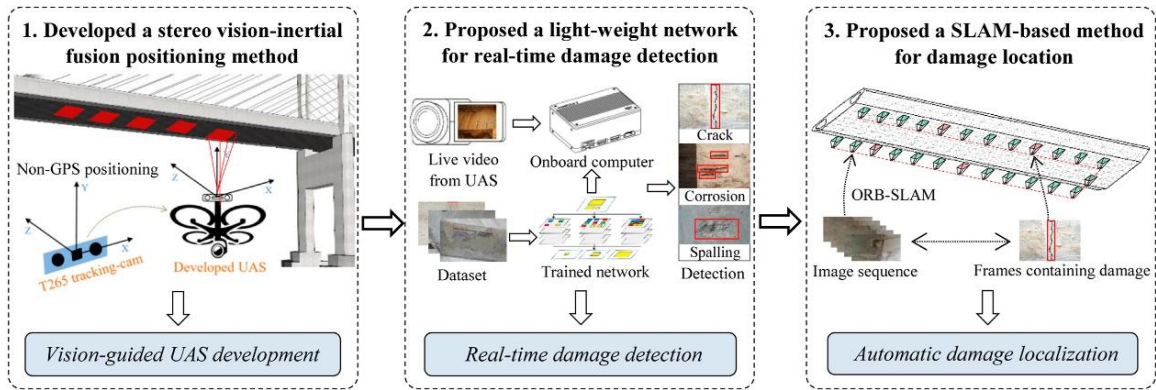


Figure 2.13 – Damage Detection and Localization Framework (Jiang et al., 2023)

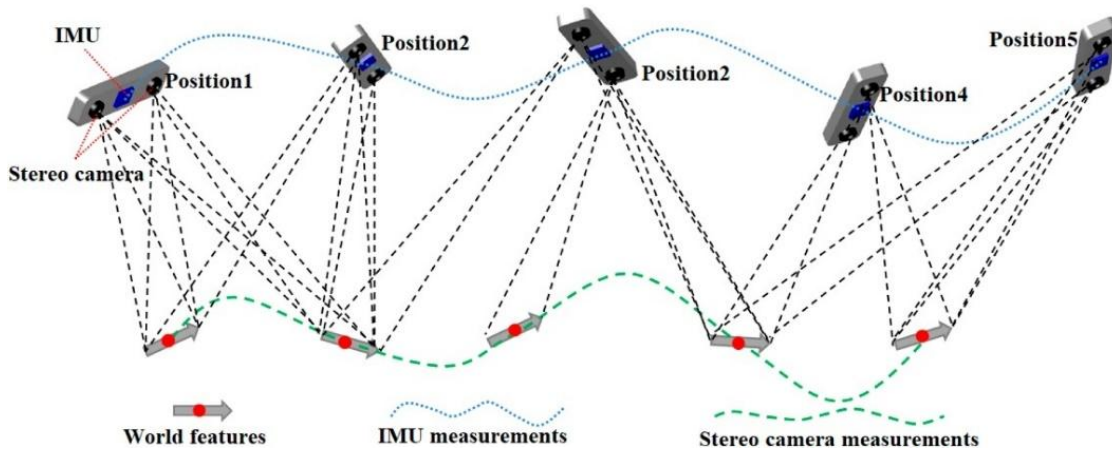


Figure 2.14 – A Damage Localization Technique by Jiang et al. (2023)

Their framework was tested on a long-span river bridge, offering a more rigorous evaluation environment than standard indoor tests due to factors like strong winds and a large detection area. Manual identification of UAV video images served as the ground truth for assessing the proposed real-time damage detection method. The results revealed a classification accuracy of 92.8% and an IoU of 80.7% between manually labeled and real-time detected damage.

Kao et al. (2023) proposed a framework to extract and quantify cracks in bridges using UAV images and deep learning-based object detection models (**Fig. 2.15**). They used planar markers to retrieve scales of images then cracks. To retrieve the crack boundaries, the authors used a local thresholding method to create a binarized image. Following that, the authors employed a crack width measurement technique in reference to the planar marker. Their YOLOv4 deep learning model was trained for crack detection and achieved an accuracy of 92%. This model performed well in identifying cracks even in images with uneven lighting and complex backgrounds. The study also demonstrated that the overall crack measurement accuracy exceeded 0.22 mm. Two edge detection methods evaluated in the study showed similar performance; however, the Canny edge detector's results varied with different thresholds, leading to greater discrepancies between measured and actual crack widths. In contrast, the morphological edge detector, which does not rely on thresholds, provided crack edges more consistent with the true dimensions.

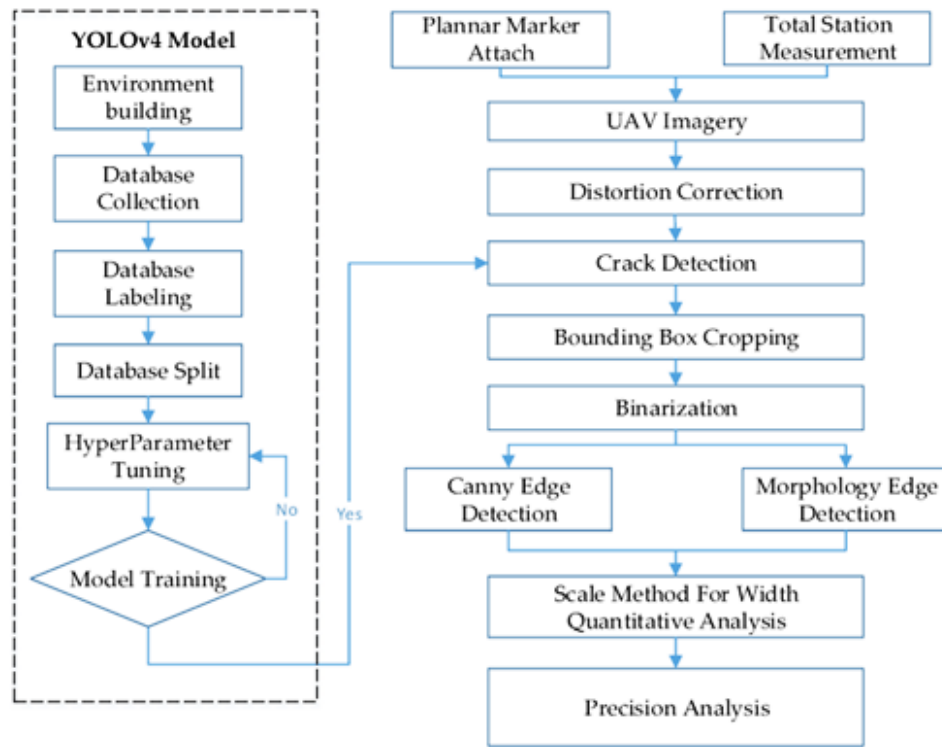


Figure 2.15 – Crack Detection and Quantification Framework (Kao et al., 2023)

Mirzazade et al. (2023) used a 3D reconstruction approach (**Fig. 2.16**) using SfM to detect and segment bridge damages based on U-Net, a popular deep learning model for semantic segmentation. As discussed earlier, SfM is a computer vision technique that reconstructs 3D structures from 2D image (only width and height) sequences taken from different viewpoints. It estimates camera positions and the 3D coordinates of scene points by analyzing the motion and structure captured in the overlapping images. They also collected Light Detection and Ranging (lidar) scans of structures to verify the integrity of the 3D reconstruction using SfM.

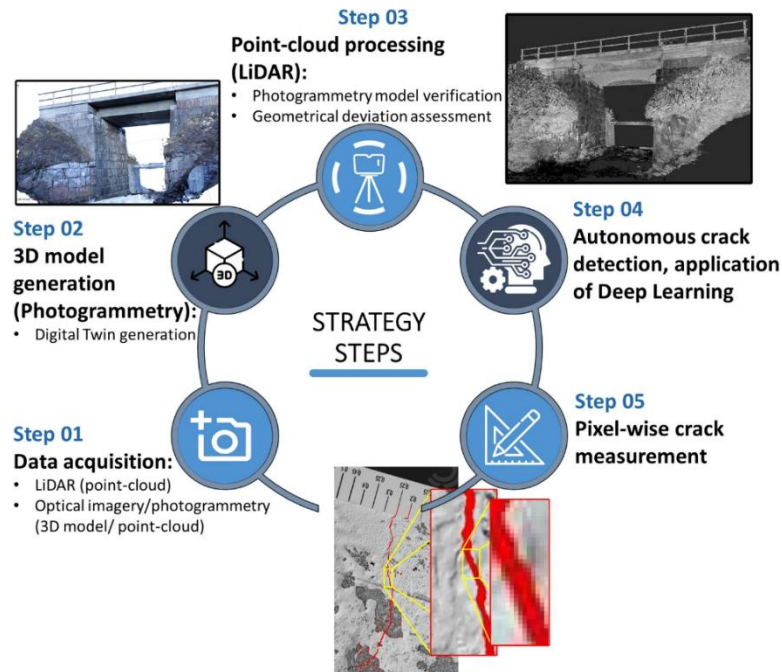


Figure 2.16 – 3D Reconstruction and Crack Detection Framework (Mirzazade et al., 2023)

Overall, the literature indicates that computer vision is a viable technique to detect and quantify many of bridge defects using a variety of imaging methods such as RGB, thermal, and lidar. These capabilities can be further expanded when paired with drone-based data collection, which improves coverage, efficiency, and access to hard-to-reach components.

CHAPTER 3 – QUANTITATIVE CONDITION STATES FOR SELECT BRIDGE ELEMENTS

3.1 Introduction

The AASHTO Manual for Bridge Elements Inspection (MBEI, 2019) provides four-level condition states for bridge elements. Other federal and state guidelines exist; however, most transportation agencies are moving toward using the AASHTO manual for consistency and reporting to the federal government.

As discussed in the previous chapters, this study is focused on concrete bridge decks and steel girders and how their defect detection and reporting can be expediated using new technologies. In this chapter, the AASHTO specified condition states for concrete decks and steel girders are presented first then their definitions are modified to be quantitative best suited for computer programming.

3.2 AASHTO Condition States for Concrete Decks

Table 3.1 presents different damage types seen for concrete bridge decks and includes their four-state condition definitions. Except delamination, other deck defects are on the surface and visible. Nevertheless, concrete delamination is a subsurface damage, which can only be detected using a nondestructive method or concrete removal. Delamination is a state of separation of layers within concrete and may lead to a loss of structural integrity for bridge decks and an increase in maintenance costs.

Table 3.1 – AASHTO Condition States for Concrete Decks (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination/ Spall/ Patched Area (1080)	None.	Delaminated. Spall 1 in. or less deep or 6 in. or less in diameter. Patched area that is sound.	Spall greater than 1 in. deep or greater than 6 in. diameter. Patched area that is unsound or showing distress. Does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Exposed Rebar (1090)	None.	Present without measurable section loss.	Present with measurable section loss but does not warrant structural review.	
Efflorescence/ Rust Staining (1120)	None.	Surface white without build-up or leaching without rust staining.	Heavy build-up with rust staining.	
Cracking (RC) (1130)	Insignificant cracks or moderate width cracks that have been sealed.	Unsealed moderate width cracks or unsealed moderate pattern (map) cracking.	Wide cracks or heavy pattern (map) cracking.	
Abrasion/ Wear (PSC/RC) (1190)	No abrasion or wearing.	Abrasion or wearing has exposed coarse aggregate but the aggregate remains secure in the concrete.	Coarse aggregate is loose or has popped out of the concrete matrix due to abrasion or wear.	

It is clear that the AASHTO definitions of condition states for concrete decks are mostly qualitative with no specific definitions for condition state 4. Alternative and measurable definitions are required, which can be used in computer programming.

3.3 Proposed Condition States for Concrete Decks

As discussed above, any defects that are expected to be found using computer tools need to be quantitative not qualitative. To this end, **Table 3.2** presents alternative definitions of condition states for concrete decks specifically for delamination, spalling, and patching. The new definitions are measurable and programmable. Furthermore, to minimize variations from AASHTO, the proposed definitions are kept at four levels. Very small delaminated regions are usually hard to find in practice, thus a limit of 7 in² (45 cm²) was proposed for CS-1, which is indeed insignificant compared with the area of bridge decks. The proposed area for CS-2 follows that of the AASHTO manual at this level. Based on the inspection of more than 50 bridges with experienced inspectors, the research team noticed that a 12 in. (30 cm) by 12 in. (30 cm) area was the smallest region inspectors annotated as delamination. With a 30% reduction, the research team proposed a cap of 110 in² (710 cm²) in CS-3. Any area greater than that is marked as CS-4. For cracking, due to lack of specifications, the proposed crack length for each condition state was based on an equivalent perimeter of its corresponding area for delamination. It is understood that these limits are subjective; however, the user of the inspection software (**Ch. 8**) can adjust them as needed.

Furthermore, compared with the AASHTO definitions, some parameters, such as the depth of defects, were excluded in the new definitions since they are challenging to estimate using computer vision techniques. For example, there are usually large errors in the direction perpendicular to the plane of view of cameras thus the depth estimate is less accurate using computer vision methods. Note that other types of defects were not quantified herein since they were beyond the scope of this project.

Table 3.2 – Proposed Condition States for Delamination, Spalling, Patching, and Cracking in Concrete Decks

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Delamination, Spall, or Patched Area (1080)	Delaminated area is less than or equal to 7 in ² (45 cm ²)	Delaminated area is greater than 7 in ² (45 cm ²) but less than 30 in ² (195 cm ²)	Delaminated area is greater than 30 in ² (195 cm ²) but less than 110 in ² (710 cm ²)	Delaminated area is greater than 110 in ² (710 cm ²)
Cracking (RC) (1130)	Crack length is less than 10 in. (25.4 cm)	Crack length is greater than 10 in. (25.4 cm) but less than 20 in. (50.8 cm)	Crack length is greater than 20 in. (50.8 cm) but less than 40 in. (101.6 cm)	Crack length is greater than 40 in. (101.6 cm)

3.4 AASHTO Condition States for Steel Girders

Table 3.3 presents different damage types seen for steel elements and includes their four-state condition definitions. Except scour, other defects are usually present on steel girders. **Table 3.4** presents the condition states for the steel protective coating.

Table 3.3 – AASHTO Condition States for Steel Elements (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Corrosion (1000)	None.	Freckled rust. Corrosion of the steel has initiated.	Section loss is evident, or pack rust is present but does not warrant structural review.	The condition warrants a structural review to determine the effect on strength or serviceability of the element or bridge; OR a structural review has been completed and the defects impact strength or serviceability of the element or bridge.
Cracking (1010)	None.	Crack that has self-arrested or has been arrested with effective arrest holes, doubling plates, or similar.	Identified crack that is not arrested but does not warrant structural review.	
Connection (1020)	Connection is in place and functioning as intended.	Loose fasteners or pack rust without distortion is present but the connection is in place and functioning as intended.	Missing bolts, rivets, or fasteners; broken welds; or pack rust with distortion but does not warrant a structural review.	
Distortion (1900)	None.	Distortion not requiring mitigation or mitigated distortion.	Distortion that requires mitigation that has not been addressed but does not warrant structural review.	
Settlement (4000)	None.	Exists within tolerable limits or arrested with no observed structural distress.	Exceeds tolerable limits but does not warrant structural review.	
Scour (6000)	None.	Exists within tolerable limits or has been arrested with effective countermeasures.	Exceeds tolerable limits but is less than the critical limits determined by scour evaluation and does not warrant structural review.	The element has impact damage. The specific damage caused by the impact has been captured in CS 4 under the appropriate material defect entry.
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in CS 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in CS 3 under the appropriate material defect entry.	

Table 3.4 – AASHTO Condition States for Steel Protective Coating (AASHTO MBEI, 2019)

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Chalking (Steel Protective Coatings) (3410)	None.	Surface dulling.	Loss of pigment.	Not applicable.
Peeling/Bubbling/Cracking (Steel Protective Coatings) (3420)	None.	Finish coats only.	Finish and primer coats.	Exposure of bare metal.
Oxide Film Degradation Color/Texture Adherence (Steel Protective Coatings) (3430)	Yellow-orange or light brown for early development. Chocolate-brown to purple-brown for fully developed. Tightly adhered, capable of withstanding hammering or vigorous wire brushing.	Granular texture.	Small flakes, less than 1/2-in. diameter.	Dark black color. Large flakes, 1/2-in. diameter or greater, or laminar sheets or nodules
Effectiveness (Steel Protective Coatings) (3440)	Fully effective.	Substantially effective.	Limited effectiveness.	Failed; no protection of the underlying metal.
Damage (7000)	Not applicable.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 2 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 3 under the appropriate material defect entry.	The element has impact damage. The specific damage caused by the impact has been captured in Condition State 4 under the appropriate material defect entry.

Again, it is clear that the AASHTO definitions of condition states for steel girders are mostly qualitative with no specific definitions for condition state 4. Alternative and measurable definitions are required for computer programming.

3.5 Proposed Condition States for Steel Girders

As discussed in the previous section, the AASHTO based condition states are qualitative. For implementation of these condition states using computer vision tools, quantitative values for four condition states of different defects are needed. **Table 3.5** presents the proposed condition classification for steel girders. In this study, a defect with an area equal to or smaller than 0.3% of the total surface area of the girder was selected as CS-1. For CS-2, the area of defects has a range of 0.3 to 3% of the girder's total surface area. For CS-3, the area of defects has a range of 3 to 16% of the girder total surface area. If the area of the defect is more than 16% of the girder's total surface area, then it falls under the category of CS-4.

These proposed condition states for steel girders were selected based on a report by Bowman et al. (2022). Note that the crack in steel girders is very critical. A small crack can lead to a catastrophic failure. A fracture critical member inspection may be carried out to inspect the critical girders, and appropriate measures should be taken to ensure safety of steel bridges. Ohio DOT (2014) recommends a CS of 2 for crack lengths up to 2 in., and Wisconsin DOT (2024) proposes a CS of 4 for crack lengths greater than 3 in.

Table 3.5 – Proposed Condition States for Steel Girders

Defects	CS 1	CS 2	CS 3	CS 4
	Good	Fair	Poor	Severe
Corrosion/Paint loss (1000)	Corrosion/paint loss less than 0.3% of total surface area of the girder	Corrosion/paint loss greater than 0.3% and less than 3%	Corrosion/paint loss greater than 3% and less than 16%	Corrosion/paint loss greater than 16%
Cracking (1010)	Crack length less than 0.25 in.	Crack length greater than 0.25 in. but less than 2 in.	Crack length greater than 2 in. but less than 3 in.	Crack length greater than 3 in.
Chalking (Steel Protective Coatings) (3410)	Chalking less than 0.3% of total surface area of the girder	Chalking loss greater than 0.3% and less than 3%	Chalking greater than 3% and less than 16%	Chalking greater than 16%
Peeling/Bubbling (Steel Protective Coatings) (3420)	Peeling/Bubbling less than 0.3% of total surface area of the girder	Peeling/Bubbling greater than 0.3% and less than 3%	Peeling/Bubbling greater than 3% and less than 16%	Peeling/Bubbling greater than 16%

CHAPTER 4 – BRIDGE DECK AND STEEL GIRDER INSPECTION DATASETS

4.1 Introduction

Large data is needed for successful training of neural networks that are sufficiently accurate in performing the desired tasks. For example, if crack detection in images is desired, thousands and sometimes millions of images with and without cracks are required. Most civil engineering related deep learning studies do not have access to such large data; however, data augmentation approaches and pre-trained networks can be utilized to overcome such data scarcity. A handful of datasets targeting specific bridge elements have been compiled in some recent studies. These datasets include mono-modality, such as Red-Green-Blue (RGB) images or thermal images, and multimodality such as RGB and thermal images combined. In this chapter, a few key past studies collecting bridge related data are reviewed first. Subsequently, the inspection data collected in the present study is discussed.

4.2 Existing Image Datasets

There are several existing datasets specific to bridge health assessments including bridge elements such as columns, decks, and overlays. **Table 4.1** presents a summary of past studies related to image datasets. Specifically designed for crack detection on concrete surfaces, the CrackForest dataset by Shi et al. (2016) includes annotated images of cracks on various concrete structures, including bridges. A derivative of the Common Objects in Context (COCO) dataset by Bianchi et al. (2020), was tailored for bridge inspection tasks, which includes annotated images of bridges highlighting various structural elements and potential defects. Dorafshan et al. (2021) provided a dataset containing images of concrete surfaces, including bridges, annotated for different types of defects such as cracks and spalling. The Road Damage Detection (RDD) dataset by Arya et al. (2022) is mostly for pavement defects but it also includes images of bridge decks.

Table 4.1 – Summary of Past Studies on Bridge Related Image Datasets

Reference	Dataset	Categories	Data Type	Size	Application
Shi et al. (2016)	CrackForest	Cracks	RGB Image	10,000	Detect and analyze cracks on the road surface
Zhang et al. (2016)	CCIC	Cracks	RGB Image	40,000	Detect cracks on concrete surfaces
Rubio et al. (2019)	--	Exposed Rebar, Delamination	RGB Images	734	Detect and quantify exposed rebars and delamination
Zou et al. (2019)	DeepCrack	Cracks	RGB Image	35,100	Detect fine cracks on various road surfaces
Bianchi et al. (2020)	COCO Bridge Damage	Bearing, Cover-plates, Gusset-plates, Stiffeners	RGB Image	774	Detect bearing, cover-plate, gusset-plate, stiffeners and evaluate their conditions
Dorafshan et al. (2021)	SDNET2021	Delamination	Infrared, Impact Echo, Ground Penetrating Radar	936	Detect delamination and reinforcement corrosion
Arya et al. (2022)	RDD2022	Cracks and Potholes	RGB Image	47,420	Detect and classify road damage types, such as cracks, potholes, and surface wear
Tazarv et al. (2022)	RC Column Damage	Cracks, Spalling, Exposed Rebars	RGB Image	3,036	Detect post-earthquake damages of RC bridge columns and determine their damage states

The available datasets do not explicitly include deck delamination and steel girder defect data, which is crucial for training and validating models in the present study. Furthermore, the current datasets lack thermal image data, which can be used for identifying internal defects like delamination.

Thermal imaging provides a unique perspective by highlighting temperature variations that may indicate underlying structural issues. For instance, delaminated areas often exhibit different thermal properties compared to intact regions, making thermal images a powerful tool for nondestructive evaluation (NDE). Further, the existing datasets do not include annotations from experts, such as the Department of Transportation (DOT) inspectors, for delamination to be used as ground truth. Expert annotations are essential for creating reliable and accurate training datasets. Inspector annotations serve as the benchmark for evaluating the performance of models and ensuring that the models are learning to identify defects correctly.

In summary, collecting new bridge inspection data was essential in the present study due to the abovementioned limitations in the existing datasets (lacking inspector annotated delamination data, steel girder defect data, and thermal data).

4.3 New Image Datasets for Concrete Bridge Decks

To collect new data for bridge decks, the research team accompanied by inspectors from the South Dakota Department of Transportation (SDDOT) inspected more than 50 bridges in South Dakota in the summers of 2023 and 2024. The inspectors used the chain-drag method, an NDE technique, to evaluate bridge decks for delamination and annotated them with paints. The research team then collected RGB images using digital single-lens reflex (DSLR) cameras, iPhone 14 Pro, and DJI drones (Matrice 200, and Mavic 3T) before and after the annotations. Furthermore, PolyCam, a mobile application utilizing iPhone lidar camera, was used to scan the bridge decks to record three-dimensional (3D) meshes of bridge decks.

RGB images taken by iPhone 14 Pro had a resolution of 8064 by 6048 pixels. Further, the RGB images obtained by the DJI Mavic 3T and the DSLR camera had a resolution of 4000 by 3000 and 4608 by 3456 pixels, respectively. The images taken by the drones were taken approximately at a height of 10 ft (3 meters) from the deck surface. For the images taken by the other cameras, the height varied between 3 to 6 ft (1 to 2 meters). **Figure 4.1** shows the data collection for iPhone-based lidar, and drone based RGB and thermal data. **Figure 4.2** shows a few sample images from each of these devices. More than 13,000 RGB and 2,000 thermal images were gathered from 50 bridges. Furthermore, the deck of 18 bridges with delamination were lidar scanned and data was processed per bridge. Additional inspection data was also provided by DOF&PF and was used in the AI model training.

PolyCam utilizes the integration of lidar data and photogrammetry to enhance the precision and detail of 3D models generated using iPhones/iPads that are equipped with lidar sensors, such as the iPhone 12 Pro and subsequent models. The process initiates with the iPhone's lidar sensor, which emits light pulses to measure the time taken for the reflections to return from various objects, thereby generating a highly accurate depth map of the scene. Concurrently, users capture multiple images or videos of the object or scene from various angles using the iPhone's camera. Photogrammetry algorithms subsequently analyze these visual inputs to discern the

structural, textural, and detailed attributes of the scene. The integrated data then undergoes further processing to produce a coherent 3D model (point cloud) of the target object.

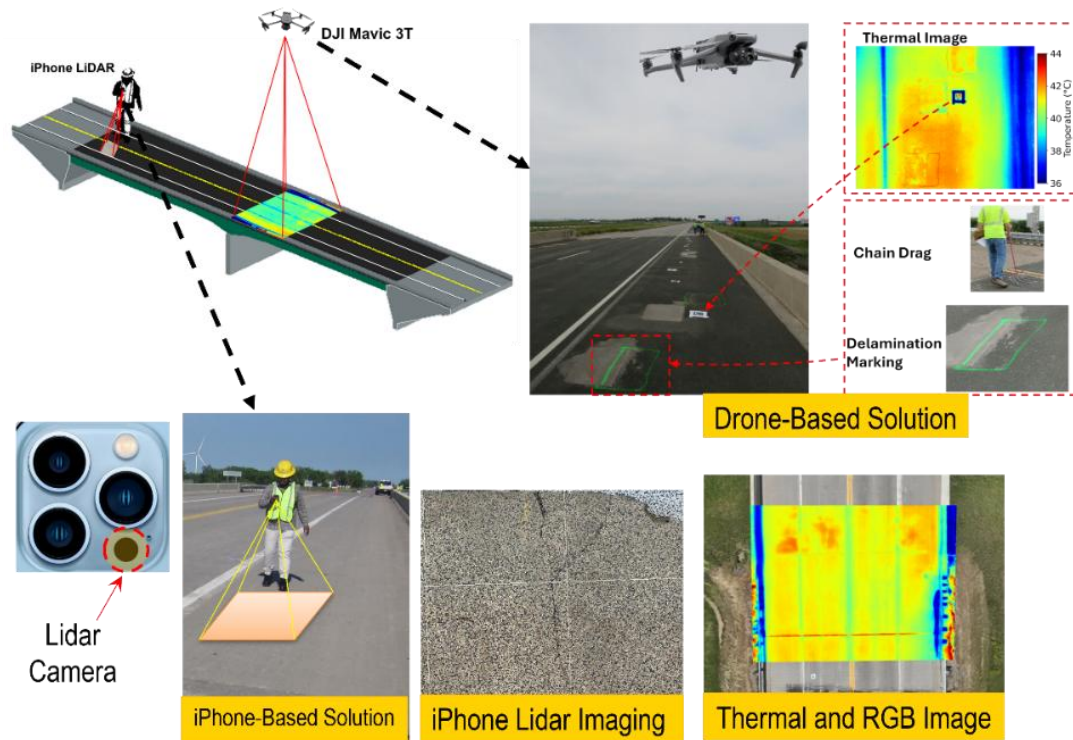
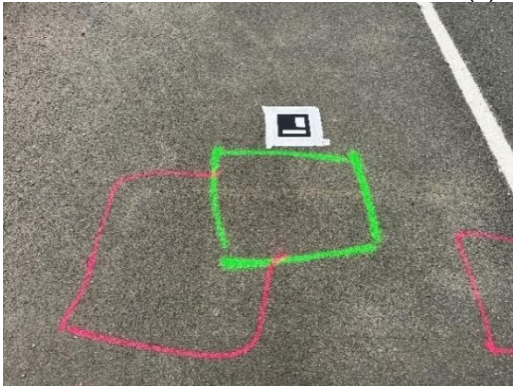


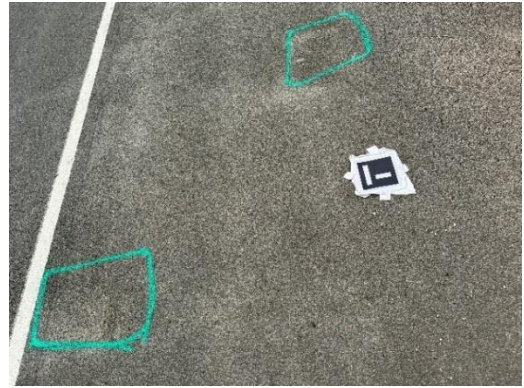
Figure 4.1 – Data Collection using iPhone's Lidar and Drone's Thermal and RGB Cameras



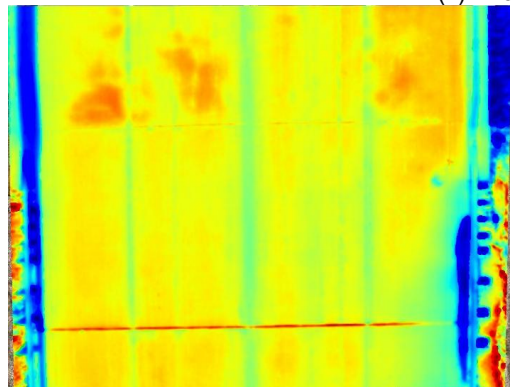
(a) DSLR Camera



(b) iPhone 14 Pro



(c) DJI Mavic 3T (RGB)



(d) DJI Mavic 3T (Thermal)

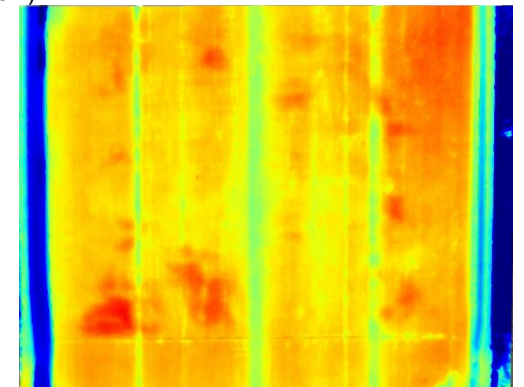


Figure 4.2 – Sample RGB and Thermal Images from Various Devices Including Inspector Annotations

4.4 New Image Dataset for Bridge Steel Girders

For steel girders, images were collected using handheld cameras and drones (e.g., **Fig. 4.3**). More than 6,700 images of steel girders were collected. Note that most of the observed defects in steel girders were the loss of protective paints.



Figure 4.3 – Sample RGB Images for Steel Girders

CHAPTER 5 – RGB BASED AI DELAMINATION DETECTION

5.1 Introduction

Computer vision based civil infrastructure inspection tools have been developed over the years to automate bridge inspection tasks such as detection of cracks, spalling, and bar corrosion (Koch et. al., 2015). Some studies were focused on a specific damage type (e.g., crack detection of bridge components, Prasanna et al., 2012) and a few explored multiple deficiencies (e.g., Oh et al., 2009; Lee et al., 2014) to produce more general inspection tools. The recent advancement in image processing and deep learning provides new pathways to design and deploy cost effective and efficient inspection tools than conventional inspections, which are mostly manual. Especially, solving computer vision tasks using deep convolutional neural networks (DCNN) has gained superior traction in recent years.

In the present project, the research team developed a computer vision based bridge defect detection and condition state estimation software (Br.CVIT, discussed later in **Ch. 8**) that was focused on delamination of concrete bridge decks and paint loss of steel girders. The present chapter discusses the development of delamination detection tools from deck RGB maps (formed from either the smartphone's lidar data or drone's RGB data), **Ch. 6** presents the proposed methods and techniques of delamination detection from deck thermal maps (formed from drone's RGB-thermal data), and **Ch. 7** offers a human-in-the-loop method of defect quantification for steel girders.

5.2 Three Pipelines for Bridge Deck Condition Assessments

Figure 5.1 shows a high-level architecture of the software for bridge decks, which has several interconnected modules to allow data collection and analyses using different devices and methods. The three main pipelines for bridge deck condition assessments include lidar-based inspection, RGB-based inspection, and thermal-based inspection.

In the lidar-based method, 3D point-cloud maps of bridge decks are obtained using iPhone/iPad's lidar camera. The point-cloud data is then run through the software to generate a 2D map of bridge decks. The software then detects local defects (delamination) using a trained neural network and pinpoints the damage on the global map of bridge decks.

In the RGB-based method, RGB images can be acquired using different sources (hand-held cameras, smartphones'/tablets' cameras, or cameras on drones). However, a drone-based data collection is recommended and built-in to develop a full 2D map of bridge decks. Subsequently, a deep learning model trained on a large image dataset is used to detect inspector annotated delamination.

The third pipeline was developed to generate 2D thermal maps of bridge decks using drone-based thermal imaging then another deep learning model detected the deck delamination in thermal maps.

The findings of all pipelines are then quantified based on the proposed damage state definitions discussed in **Ch. 3** (as default values, user can change them) and a report is generated with standard outputs and formats.

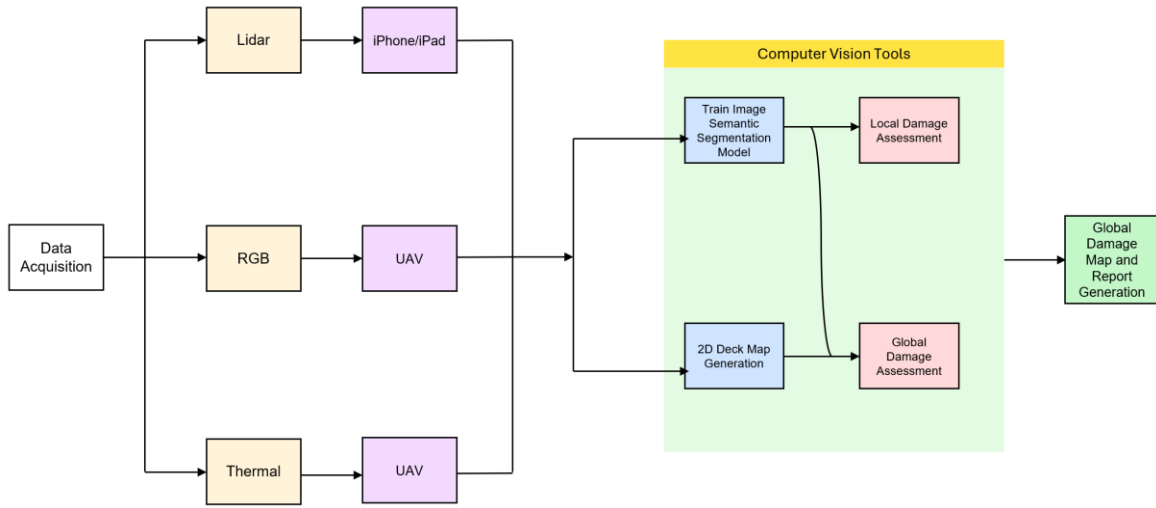


Figure 5.1 – Proposed Computer Vision Framework for Delamination Detection of Bridge Decks

In the following sections, a brief background of computer vision methods and techniques that are needed in the present study is discussed first. Developed in the present project, bridge deck RGB map generation using smartphones' lidar sensors and drones' RGB cameras is discussed next. The subsequent sections of the chapter discuss the development of the AI models and their accuracies.

5.3 Computer Vision Background

5.3.1 Deep Convolutional Neural Networks (DCNNs)

DCNNs, often simply referred to as convolutional neural networks (CNNs), have revolutionized the field of computer vision and become the cornerstone of many modern image recognition and processing systems. DCNNs are often used as latent feature extractors that are more robust than handcrafted filters for feature extraction. Convolutional layers in deep networks can learn hierarchical features from low-level edges to high-level object parts. This multi-scale feature extraction is crucial for semantic segmentation, as it helps in identifying both small and large objects in an image. Architectures like U-Net (Ronnenberger et al., 2015) use encoder-decoder structures to capture fine-grained details. The encoder compresses the image into a lower-dimensional feature space, and the decoder reconstructs the spatial details from these features. This section provides details on the building blocks of encoders in deep convolutional networks.

The concept of convolutional networks dates to the late 1980s and early 1990s with the work of LeCun et al. (1989), who developed the LeNet architecture for handwritten digit recognition (LeCun et al., 1998). However, the true potential of CNNs was realized in 2012 when Krizhevsky et al. (2012) introduced the AlexNet architecture, which significantly outperformed previous methods in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This breakthrough demonstrated the power of deep learning and large-scale datasets, sparking widespread interest and research in the area.

Convolutional layers are the fundamental building blocks of CNNs, designed to process grid-like data such as images. The primary function of a convolutional layer is to detect local patterns within an image, such as edges, textures, or more complex features, by applying a set of learnable filters (also known as kernels) across the input data.

Each filter in a convolutional layer slide over the input image to perform a dot product between the filter's weights and a local region of the input image. This operation is mathematically expressed as:

$$(X * W)_{i,j} = \sum_m \sum_n X_{i+m,j+n} W_{m,n} + b \quad (5.1)$$

where, X represents the input image, W denotes the filter, i and j are the coordinates of the output feature map, m and n iterate over the filter dimensions, and b is the bias term. The result of this dot product is a single value that forms part of the output feature map, capturing the presence of the pattern encoded by the filter at that specific location.

The convolution operation is repeated as the filter moves across the entire input image, producing a 2D activation map (feature map) for each filter. The depth of the output from a convolutional layer is determined by the number of filters used, with each filter detecting different features within the image.

Pooling layers, typically placed after convolutional layers, are used to reduce the spatial dimensions of the feature maps while retaining the most important information. This process, known as downsampling or subsampling, helps to decrease the computational load, memory usage, and overfitting.

A common type of pooling operation is max pooling, which selects the maximum value within a defined window of the input feature map. The mathematical operation for max pooling can be described as:

$$Y_{i,j} = \max_{0 \leq m < p, 0 \leq n < q} (X_{s \cdot i + m, s \cdot j + n}) \quad (5.2)$$

where $Y_{i,j}$ is the output of the pooling layer, X is the input feature map, $p \times q$ is the size of the pooling window and s is the stride, indicating the step size with which the pooling window moves across the input feature map.

Convolutional and pooling layers work together in CNNs to progressively abstract higher-level features from the raw input image. Initially, convolutional layers capture low-level features such as edges and textures. As the data passes through multiple layers of convolutions and pooling, the network learns increasingly complex features, ultimately leading to high-level representations that are useful for tasks such as classification, detection, and segmentation.

For example, after the initial convolutional and pooling layers have processed an image of a cat, the network might detect edges and textures corresponding to fur patterns and contours. Deeper layers could then recognize parts of the cat such as ears, eyes, and whiskers. Finally, the fully connected layers use these high-level features to make a prediction about the presence of a cat in the image.

In conclusion, convolutional layers identify local patterns through learnable filter-based operations, while pooling layers reduce spatial dimensions and retain important information. Together, they enable CNNs to build hierarchical feature representations that are essential for effectively analyzing and interpreting image data.

5.3.2 Semantic Segmentation of Images

Semantic segmentation is a computer vision task that involves partitioning an image into semantically meaningful regions, with each pixel assigned a class label corresponding to the object or region it represents. This task is crucial for understanding the detailed structure and composition of images, making it essential for applications in autonomous driving, medical imaging, and scene understanding.

Semantic segmentation goes beyond traditional image classification and object detection by providing a dense prediction, where every pixel in an image is labeled. The goal is to create a precise and comprehensive map of the objects and regions within an image.

The introduction of fully convolutional networks (FCNs) by Long et al. (2015) marked a significant advancement in semantic segmentation. FCNs replace the fully connected layers of traditional CNNs with convolutional layers that produce spatial outputs, enabling end-to-end learning and prediction of segmentation masks.

In a fully convolutional network, the input image X is processed through several convolutional and pooling layers, resulting in a feature map. Let $f(X, W)$ represent the output of the network given the input X and parameters W . For semantic segmentation, the output is a dense prediction with the same spatial dimensions as the input image but with a depth corresponding to the number of classes. Mathematically, the output feature map $f(X, W)$ can be expressed as:

$$f(X; W) = \text{Conv}_n(\text{Pool}_{n-1}(\dots \text{Conv}_2(\text{Pool}_1(\text{Conv}_1(X_1, W_1), W_2), \dots), W_n)) \quad (5.3)$$

where, Conv_i and Pool_i denote the convolutional and pooling layers at the i -th stage, respectively, and W_i represents the weights of the i -th convolutional layer.

To obtain pixel-level predictions, the spatial resolution of the feature maps needs to be restored to match the input image. This is achieved using upsampling techniques such as deconvolution (transposed convolution) or bilinear interpolation. The deconvolution operation can be mathematically represented as:

$$Y_{i,j} = \sum_m \sum_n X_{\lfloor \frac{i+m}{s} \rfloor, \lfloor \frac{j+n}{s} \rfloor} W_{m,n} \quad (5.4)$$

where, Y is the upsampled output, X is the input feature map, W is the filter, s is the stride, and m and n iterate over the filter dimensions.

In practice, upsampling layers are combined with skip connections from earlier layers in the network to refine the segmentation results and recover spatial details lost during downsampling. This approach is exemplified in architectures like U-Net (Ronneberger et al., 2015), which uses

symmetric downsampling and upsampling paths with skip connections to merge high-resolution features from the encoder with upsampled features in the decoder.

Training a semantic segmentation model involves minimizing a loss function that measures the discrepancy between the predicted segmentation map and the ground truth. A common choice is a pixel-wise cross-entropy loss, defined as:

$$\mathcal{L}_{CE} = - \sum_{i,j} \sum_c t_{i,j,c} \log(p_{i,j,c}) \quad (5.5)$$

where, $t_{i,j,c}$ is the ground truth binary indicator for class c at pixel (i, j) , and $p_{i,j,c}$ is the predicted probability for class c at pixel (i, j) .

Another loss function that is commonly used is the Dice coefficient loss, which measures the overlap between the predicted and ground truth masks:

$$\mathcal{L}_{Dice} = 1 - \frac{2 \sum_{i,j} p_{i,j} t_{i,j}}{\sum_{i,j} p_{i,j}^2 + \sum_{i,j} t_{i,j}^2} \quad (5.6)$$

where, $p_{i,j}$ and $t_{i,j}$ are the predicted and ground truth binary values at pixel (i, j) , respectively. In the present research, the Dice loss function was used to train the models.

5.3.3 Stereovision and World Mapping

Stereovision, also known as stereoscopic vision or stereo imaging, is a technique used in computer vision and robotics to perceive depth by utilizing two or more images of the same scene taken from different viewpoints. Stereovision enables machines to reconstruct the 3D structure of their environment. **Figure 5.2** shows a typical stereo system. The fundamental principle behind stereovision is triangulation. When two cameras, separated by a known baseline distance, capture images of the same scene, corresponding points in the images can be identified. The disparity, or difference in position, of these projected 2D points between the images is inversely proportional to their distance from the cameras. By calculating the disparity, the depth information can be extracted, allowing for the reconstruction of a 2.5D (depth from a single direction) scene.

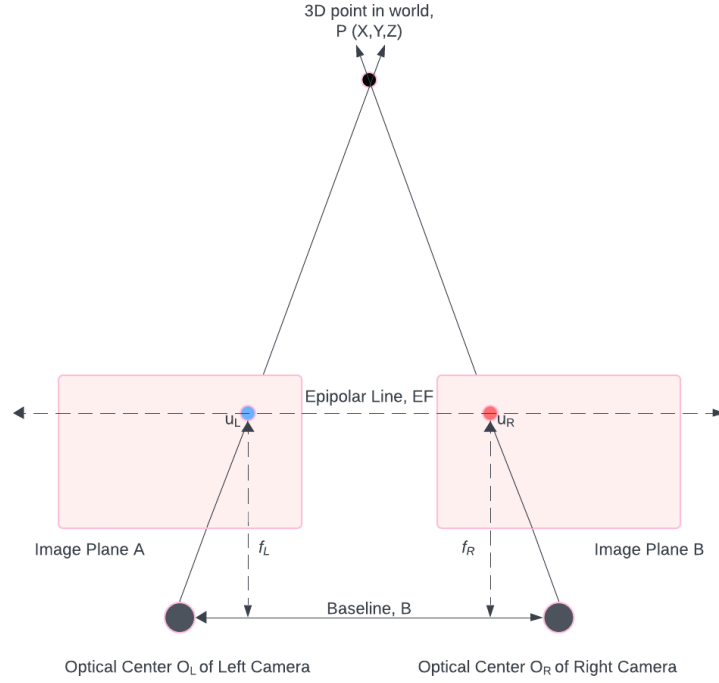


Figure 5.2 – Depth Estimation from Stereo Camera Setup

In the above figure (**Fig. 5.2**), two cameras are perfectly parallel, meaning that there is no relative rotation between the cameras. The baseline, B , between the cameras determines the distance between the optical centers of the system. f_L and f_R are the focal lengths of the left and right cameras, respectively. Both cameras are looking at a real-world point, P , with a coordinate of (X, Y, Z) . The projected image coordinate of P in the left camera is u_L and in the right camera is u_R . The projected image coordinates in the image planes can be estimated using:

$$u_L = f_L \frac{X}{Z} \quad (5.7)$$

$$u_R = f_R \frac{X - B}{Z} \quad (5.8)$$

The distance between the two projected points is known as disparity and is given by,

$$disparity = u_L - u_R = \left(\frac{f_L + f_R}{2} \right) \frac{B}{Z} \quad (5.9)$$

The depth information of the real-world point, P , which is the distance between the center of baseline and P is modeled by,

$$depth = \left(\frac{f_L + f_R}{2} \right) \frac{B}{disparity} \quad (5.10)$$

The actual process of stereovision setup requires more complex estimation because the camera lenses always induce distortions to the acquired images. Before computing the depth information, it is required to undistort both cameras' images using their distortion coefficients and intrinsic parameters. The process of estimating the camera matrix and the distortion coefficients is known as camera calibration. The projection of a real-world point to the image plane (both homogenous coordinates) is represented in terms of a linear transformation using the camera matrix as:

$$\begin{bmatrix} wu \\ wv \\ w \end{bmatrix} = K [R | T] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = P \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (5.11)$$

where, P is known as the camera matrix and is composed of the intrinsic and extrinsic parameters, K includes the intrinsic parameters, R is a 3-by-3 rotation matrix, and T is the translation vector. The intrinsic parameters contain information about the focal lengths in the X and Y directions and the coordinates of the principal point in both directions as well. The matrix is given by:

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (5.12)$$

where, f_x and f_y are the focal lengths in the X and Y directions. The principal point is given by c_x and c_y .

The most common method of estimating these parameters is through a 2D planer pattern to compute the calibration parameters, which is discussed in the next section with more details. The reason this method is popular is that it simplifies the calibration process. Since the pattern is flat and lies on a single plane (with no Z-axis components), it eliminates the complexity of dealing with depth information or 3D spatial relationships. This makes it easier to compute calibration parameters, such as intrinsic camera parameters (such as focal length and optical center) and extrinsic parameters (such as the camera's position and orientation relative to the pattern). By focusing only on the 2D plane, the process becomes less computationally intensive and more straightforward, as the calibration only needs to account for distortions and errors in the 2D image plane without additional complications from depth. So, the projection equation reduces to:

$$x_i = PX_i; \quad Z_i = 0 \quad (5.13)$$

here, X_i is the 3D point with a coordinate of $(X_i, Y_i, 0)$ and x_i is the corresponding 2D point in image plane (u_i, v_i) .

World mapping, in the context of computer vision and robotics, refers to the process of creating a comprehensive, 3D representation of an environment. This map serves as a crucial reference for navigation, planning, and interaction within the environment. World mapping often leverages

data from various sensors, including stereoscopic cameras, lidar, and GPS, to build an accurate and detailed map. Key components of a world mapping framework include:

- **Camera Calibration:** Calibrate both cameras to determine their intrinsic parameters (camera matrix, distortion coefficients) and the extrinsic parameters (rotation and translation between the cameras)
- **Image Rectification:** A technique to rectify the images to align them as if they were captured by a camera pair with parallel optical axes. This simplifies the correspondence problem to a 1D search along horizontal lines.
- **Disparity Map Generation:** Compute the disparity map, which measures the pixel-wise difference between the corresponding points in the left and right rectified images.
- **Point Cloud Generation:** Convert the disparity map to a point cloud representing the 3D coordinates of each pixel.
- **3D Reconstruction:** Construct a 3D mesh from the point cloud by connecting the points to form a surface.

Both stereovision and world mapping are pivotal technologies in the field of computer vision and robotics, providing machines with the ability to perceive, understand, and interact with their surroundings in three dimensions. By mimicking human depth perception and creating detailed maps of the environment, these technologies drive advancements in autonomy, navigation, and interaction across various industries, enhancing both functionality and safety.

5.3.4 Camera Calibration

As discussed above, in a camera calibration procedure, intrinsic and extrinsic parameters of the camera are estimated. The camera calibration helps to minimize image distortions (e.g., radial and tangential distortions) caused by the lens. It also provides accurate geometric mapping between world coordinates and image coordinates. To obtain the camera calibration parameters, multiple images of a known calibration pattern, such as a checkerboard or circular grid, are needed. A basic camera calibration equation can be written as:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (5.14)$$

where, X , Y , and Z are the coordinates of a 3D point in the world coordinate; x and y are the coordinates of the corresponding 2D point in the image; K is the intrinsic matrix, which includes the focal length and the optical center of the camera; R is the rotation matrix that describes the orientation of the camera relative to the world; and t is the translation vector that represents the position of the camera relative to the world. The combination of R and t i.e., $[R|t]$ is called an extrinsic matrix, which provides the camera's position and orientation relative to the world.

The intrinsic matrix, K , describes the camera's internal parameters including the focal length and the optical center (principal point) and can be obtained from **Eq. 5.12**.

Radial and tangential distortions are the most common lens distortion in the images:

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2p_1 xy + p_2(r^2 + 2x^2) \\ p_1(r^2 + 2y^2) + 2p_2 xy \end{bmatrix} \quad (5.15)$$

where, x_d and y_d are the distorted image coordinates; x and y are the undistorted image coordinates; $r^2 = x^2 + y^2$ is the radial distance from the image center; k_1 , k_2 , and k_3 are the radial distortion coefficients; and p_1 , and p_2 are the tangential distortion coefficients.

In the present study, several cameras were used to obtain bridge data. For example, a DJI Mavic 3T drone was used, which was equipped with a pair of RGB and thermal cameras. The resolution of the RGB image, when simultaneously taking both RGB and thermal images, was 4000×3000 pixels whereas the resolution of the thermal image was 640×512 pixels. To calibrate the thermal and RGB cameras of this drone, a grid with eight rows and columns of circles was used. **Figure 5.3** presents the raw thermal images of this grid. The raw thermal images were processed to a color version showing actual temperatures using in-house coding (**Figure 5.4**). Similarly, **Figure 5.5** shows sample RGB images including the circular grid used to calibrate the RGB camera. Note that for thermal camera calibration, the circular grid consisted of reflective circles to easily detect them in thermal images. The circles were detected in each image of the calibration dataset followed by the estimation of calibration parameters to undistort the RGB and thermal images (**Figure 5.6** and **5.7**).

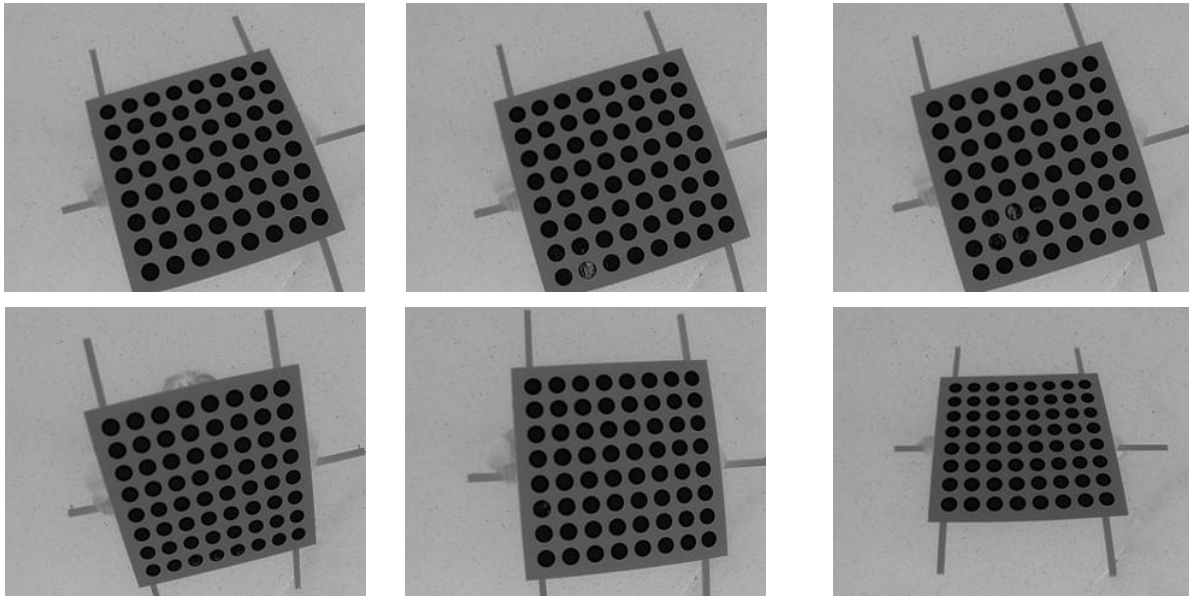


Figure 5.3 – Raw Thermal Image Dataset Collected for Thermal Camera Calibration

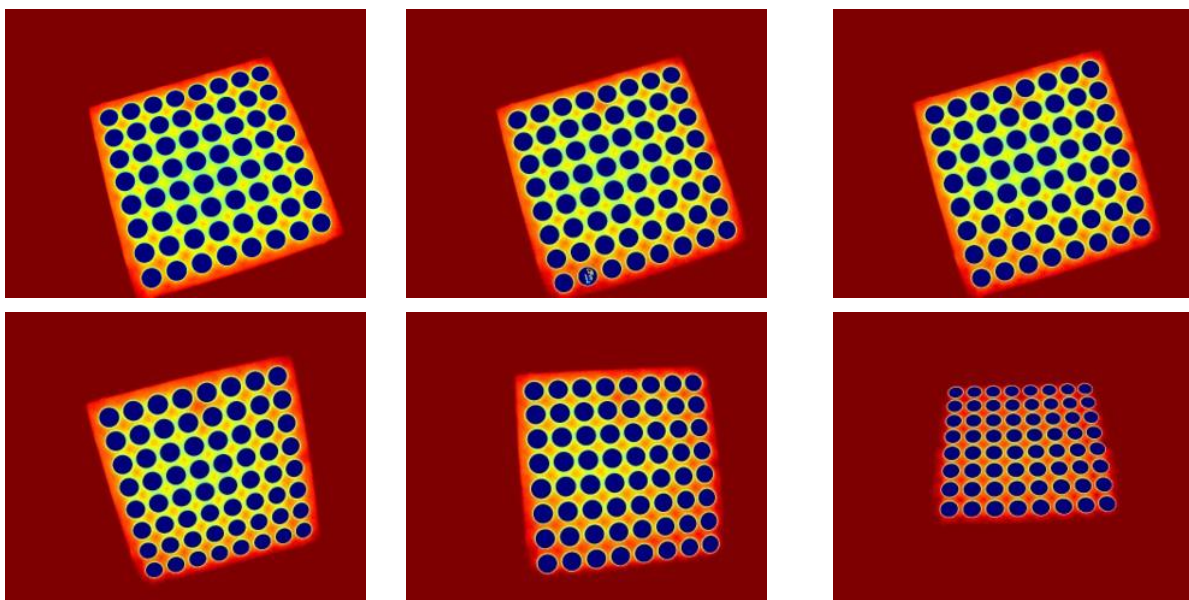


Figure 5.4 – Processed Thermal Image Dataset Collected for Thermal Camera Calibration

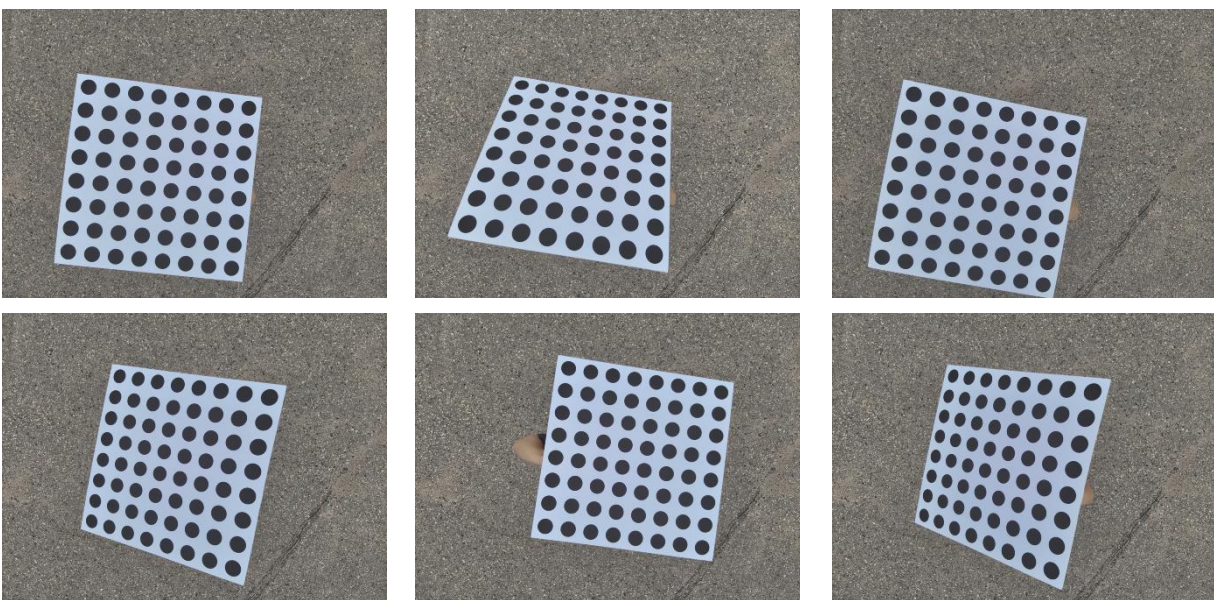


Figure 5.5 – RGB Image Dataset Collected for RGB Camera Calibration

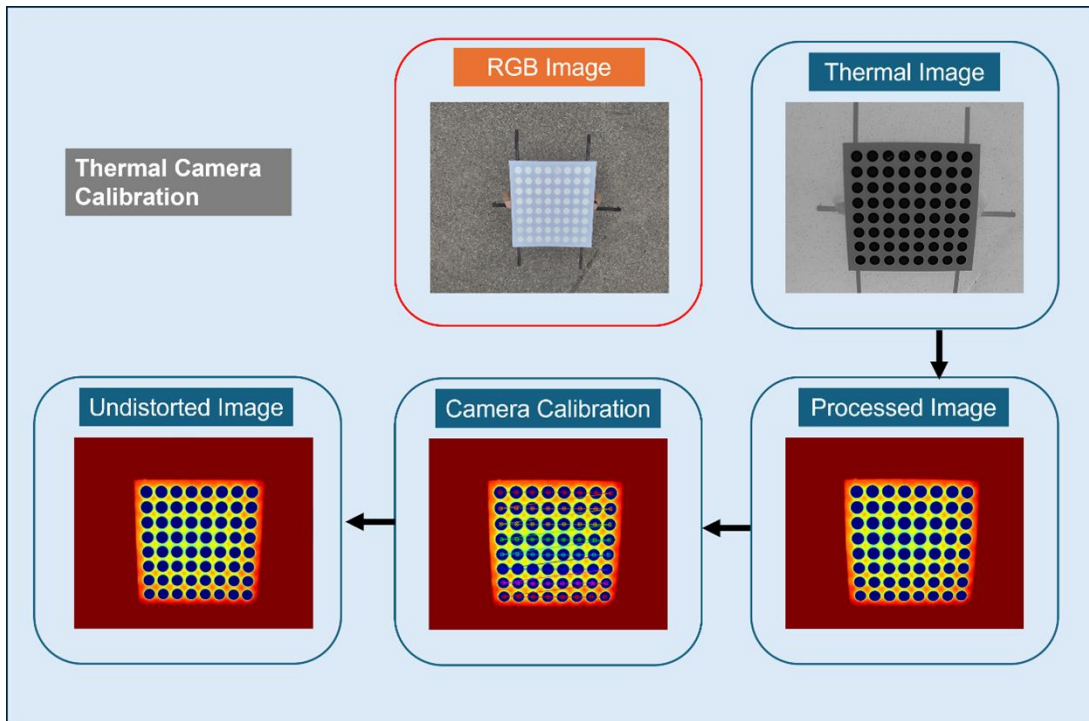


Figure 5.6 – Thermal Camera Calibration Steps

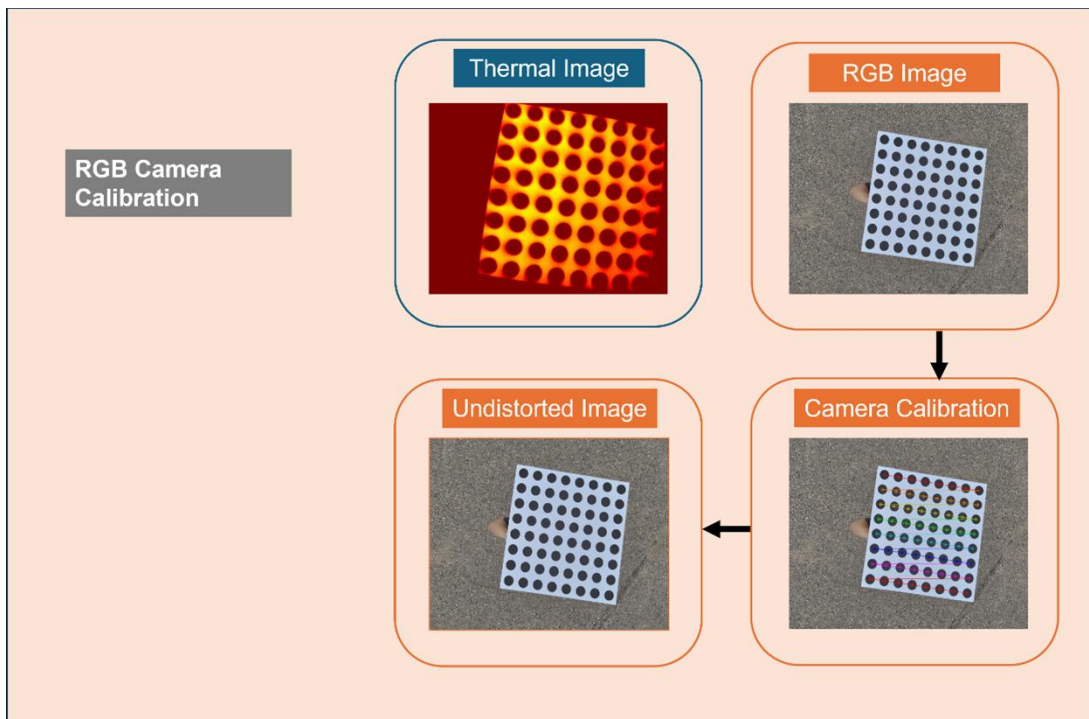


Figure 5.7 – RGB Camera Calibration Steps

Table 5.1 – Camera Calibration Parameters for Devices Used in Present Study

Parameter	RGB Drone Mavic 3T	Thermal Drone Mavic 3T	Canon PowerShot SX530 HS
Focal Length, f_x	3000.025	5733.255	3271.34
Focal Length, f_y	3000.025	5733.255	3274.61
Optical Center, c_x	1994.955	332.241	2255.56
Optical Center, c_y	1503.928	196.065	1775.56
Radial Distortion Coefficient, k_1	0.187	-19.718	-0.0321
Radial Distortion Coefficient, k_2	-0.507	193.637	0.00772
Radial Distortion Coefficient, k_3	0.28	-123.354	0.004818
Tangential Distortion Coefficient, p_1	0.001	0.178	-0.00496
Tangential Distortion Coefficient, p_2	-0.001	-0.076	0.010

5.3.5 Homography Matrix

Homography matrix transforms the coordinates of points in one image to the coordinates of corresponding points in another image as:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = H \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (5.16)$$

where, x and y are the coordinates of a point in the first image and x' and y' are the coordinates of the corresponding point in the second image. The homography matrix, H , is a 3×3 matrix:

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (5.17)$$

where, h_{11} to h_{33} are the parameters of the homography matrix that can be estimated using a set of images with corresponding points that are known. For two images, at least four corresponding points between them should be known. For the pair of points, a linear system of equations can be solved to find the homography matrix, H :

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 & -x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 & -y'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2x'_2 & -x'_2 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2y'_2 & -y'_2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_n & y_n & 1 & 0 & 0 & 0 & -x_nx'_n & -y_nx'_n & -x'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -x_ny'_n & -y_ny'_n & -y'_n \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \quad (5.18)$$

5.3.6 Feature Detection

To stitch overlapping RGB images, features must be first detected in each image and then the common features must be matched in between images. The feature detection is discussed herein, and the feature matching is discussed in the following section.

Different algorithms are available to detect features in RGB images including Harris Corner detector, Histogram of Oriented Gradients (HOG), Scale Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF), Features from Accelerated Segmentation Test (FAST), Binary Robust Independent Elementary Features (BRIEF), Oriented FAST and Rotated BRIEF (ORB), Binary Robust Invariant Scalable Keypoints (BRISK), and Accelerated KAZE (AKAZE).

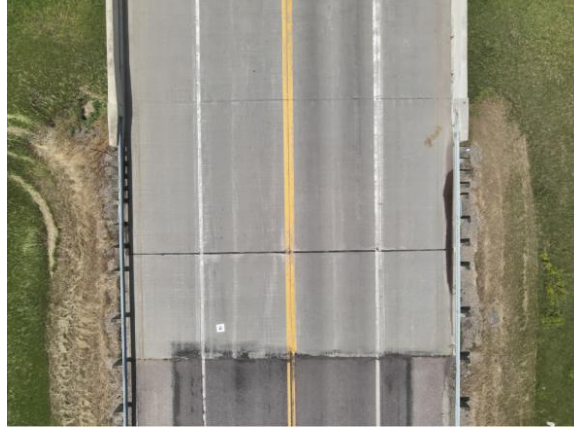
Harris Corner detector (Harris & Stephen, 1988) uses a gradient based approach to identify corners in an image. Dalal and Triggs (2005) developed the HOG algorithm to identify the human body in images based on the orientation of gradients.

SIFT (Lowe, 2004) is a robust feature detector against scale, rotation, transformation and illumination changes in images. SIFT successively blurs the original image by convolving it with a Gaussian filter then identifies the local extrema, which are potential points for the features, by comparing the current point to eight surrounding pixels in the current scale and nine pixels in higher and lower scale making a total of 26 pixels. The performance of the SIFT detector is robust compared with other detectors; however, the computational cost is its drawback.

SURF was introduced by Bay et al. (2006) which uses Hessian matrix to compute the features. The Hessian matrix is a second order derivative of the intensity values of an image. Box filter is used to estimate the Hessian matrix, which is a computationally more effective approach than the Gaussian filter used in SIFT.

FAST (Rosten et al., 2005) identifies the corner in an image by comparing a pixel to 16 other pixels in its surroundings. If there are contiguous n-number of pixels that are brighter or darker than the current pixel being tested, then the current pixel is declared as a corner. FAST is not invariant to scale and rotation i.e., any change in scale or rotation of the image will have an impact on the results. BRIEF is a feature descriptor developed by Colander et al. (2010) which efficiently stores the information pertinent to the detected feature; however, BRIEF is not invariant to rotation and scale. To improve the BRIEF descriptor, a variant called ORB was developed by Rublee et al. (2011), which is invariant to rotation. ORB combines the benefits of the FAST and BRIEF algorithms and adds the dominant orientation information for the feature using gradients which makes it invariant to rotation. Leutenegger et al. (2011) proposed BRISK algorithm to detect features in the image using different scales but uses the FAST corner detector instead of Gaussian used in SIFT. AKAZE (Alcantarilla et al., 2011) uses a nonlinear scale space based on a nonlinear diffusion to identify features at different scales instead of linear scale space used in SIFT.

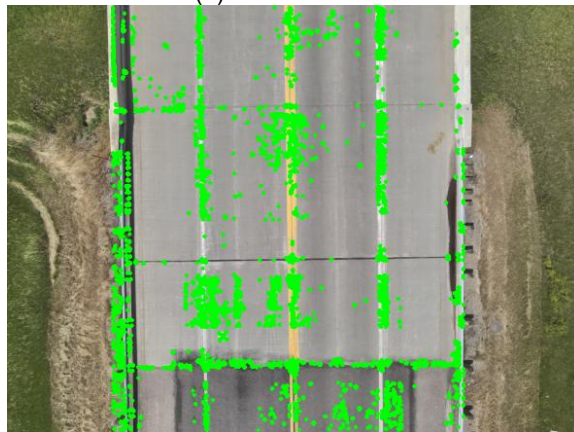
Feature detection is a crucial stage in image stitching process for bridges. **Figure 5.8** provides sample features detected in an image of a bridge deck taken by a drone using different features detector algorithms. In this study, after different trials, the SIFT detector was selected to extract features of bridge decks (**Fig. 5.9**).



(a) Original Image



(b) SIFT Detector

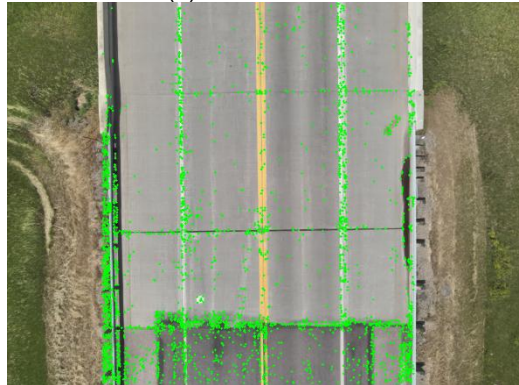


(c) Harris Corner Detector

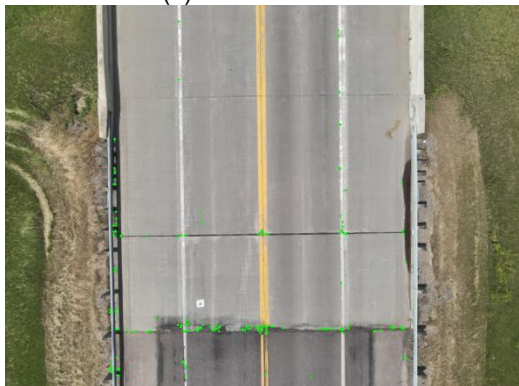
Figure 5.8 – Samples of Bridge Deck Features Detected by Different Algorithms



(d) ORB Detector



(e) FAST Detector



(f) BRISK Detector



(g) AKAZE Detector

Figure 5.8 – Continued

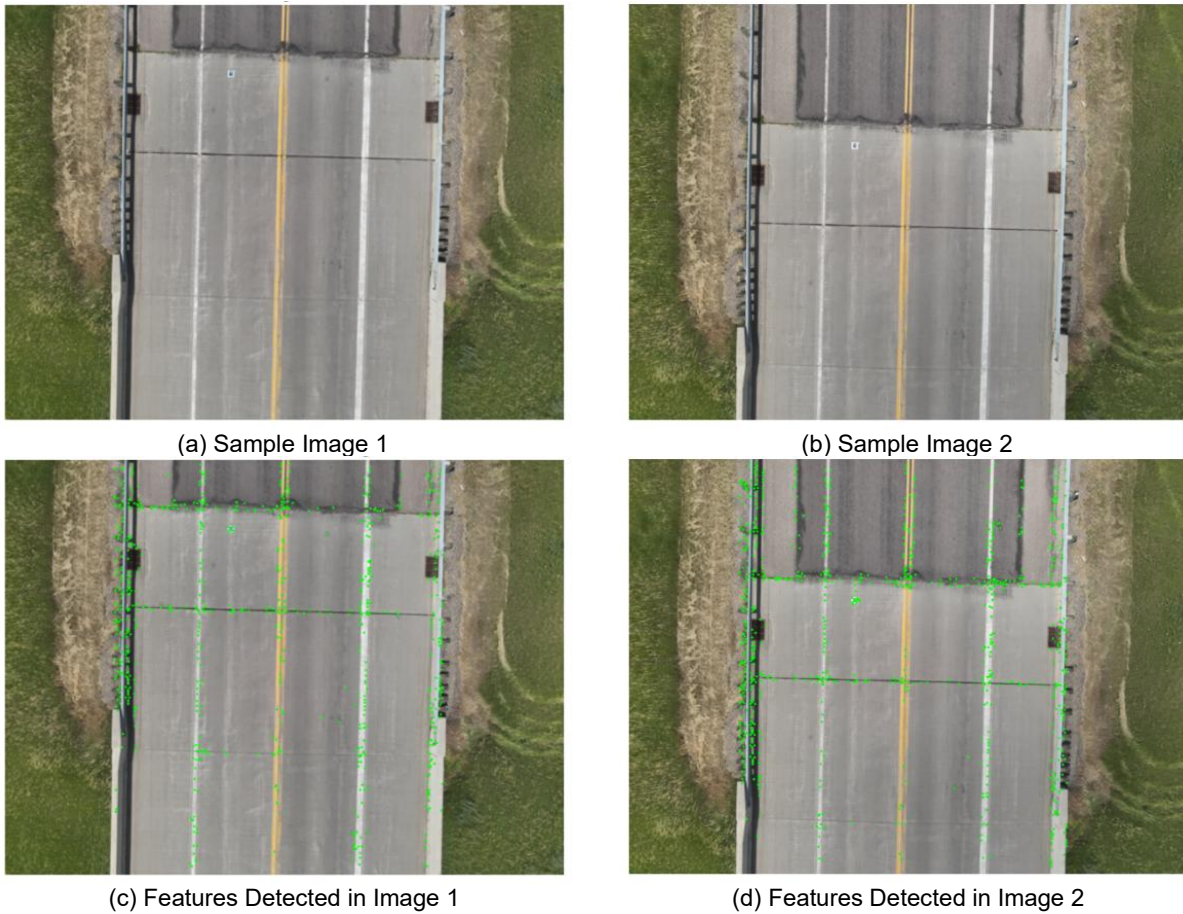


Figure 5.9 – Bridge Deck Feature Detections by SIFT Detector

5.3.7 Feature Matching

The next step after feature detection is to match the features or obtain correspondence between features in the overlapping images. Bruteforce Matcher (BFMatcher) with K-Nearest neighbors (KNN, Bradski, 2014) matching was used to identify corresponding points between a pair of images. Lowe's ratio test (Lowe, 2004) and random sample consensus (RANSAC) method (Cantzler, 1981) were used to filter out the outliers of the matched features. **Figure 5.10** shows a sample of matched features between two overlapping images of a bridge deck placed side by side. Since the two images were taken by a drone at two different locations, a portion of the images (can be as small as 25% or as large as 75%) is common between the two images. The green lines connect the best detected features of the two images. It can be seen that most common features between the two images were successfully matched (those with parallel green lines); however, some outliers (intersecting green lines) exist. To minimize the effects of outliers, only the top 100 matched features were used in this study for bridge decks.

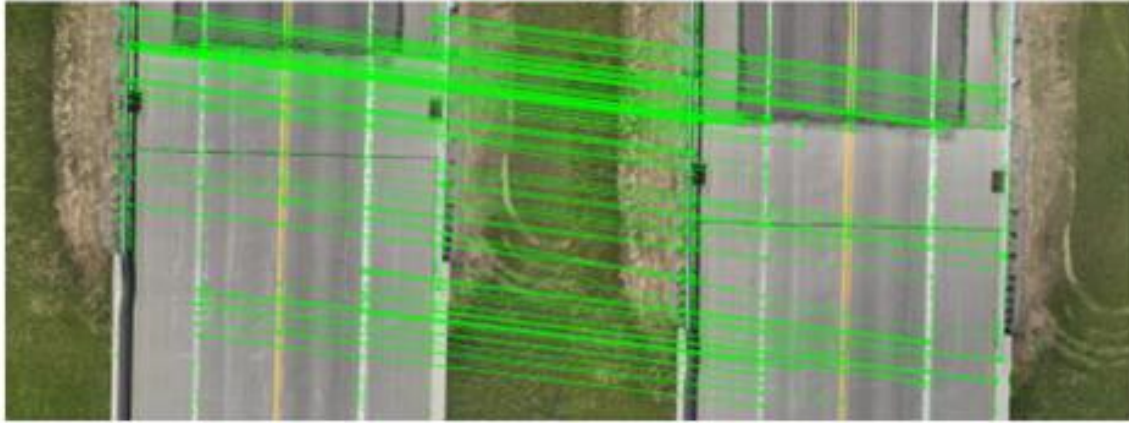


Figure 5.10 – Sample of Feature Matching for a Bridge Deck

5.4 Bridge Deck 2D RGB Map Generation

The RGB map of bridge decks (or plan view of the bridge deck from top) can be created for the entire bridge using either iPhone/iPad-based lidar data (point clouds) or drone-based RGB images. The following section provides details of 2D map generation for both approaches. Note that 2D maps are preferred over 3D objects since most AI models work better in analyzing 2D images.

5.4.1 Bridge Deck RGB Map Generation using iPhone-Based Lidar Method

In the present study, both iPhone and iPad equipped with a short-range lidar sensor were used to collect lidar data of bridge decks. Furthermore, PolyCam, a mobile application, accessed the lidar sensor and generated a point cloud (3D) data. **Figure. 5.11** shows a sample of the recorded 3D mesh in a file viewer called “MeshLab”. Note that only one lane of the bridge was scanned in this example.

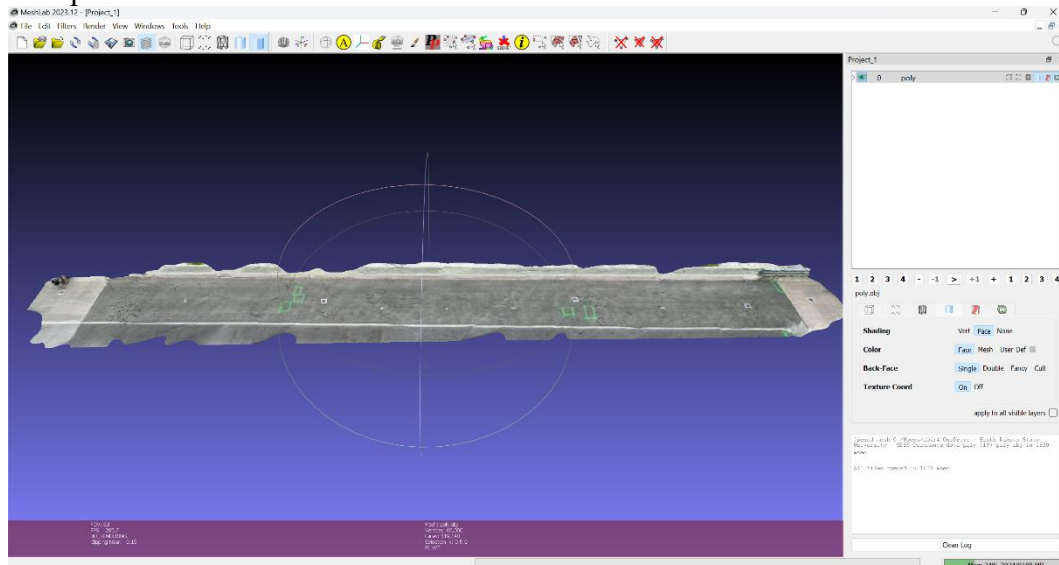


Figure 5.11 – A Sample of PolyCam Generated 3D Mesh of a Bridge Deck

Subsequently, the 3D scan recorder with an extension of OBJ including its texture map was processed using in-house programming to obtain a 2D image of the bridge deck by mapping all 3D points to a reference plane, which was the bridge deck surface in the present study. **Figure 5.12** shows a sample 2D map of a bridge deck extracted from its 3D map (**Fig. 5.11**).

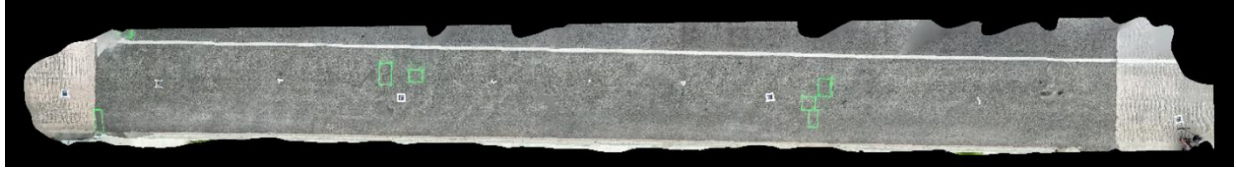


Figure 5.12 – 2D Maps of Bridge Decks Developed from Point Cloud (Mesh) Data

To obtain such a 2D image from its 3D mesh data, each of the faces of the mesh (triangles describing the planes) was processed independently to compute its projected transformations and to extract the pixel information given by the texture map. From these 3D mesh scans, an intricate process of extracting 2D maps was followed as shown in **Fig. 5.13**. The process begins with loading the mesh object exported from the PolyCam application and then parsing the object file to load the vertices, textures, and segment information. Then, the histogram of the depth of the vertices is computed to clip the vertices that are too far from the reference plane, which is the bridge deck surface. After getting vertices that fell on the deck, a Principal Component Analysis (PCA) is carried out to find the best plane that contains those vertices. Subsequently, the vertices are projected on the reference plane. Finally, the textures are extracted from the texture map given by the *uv* coordinates of the mesh descriptor.

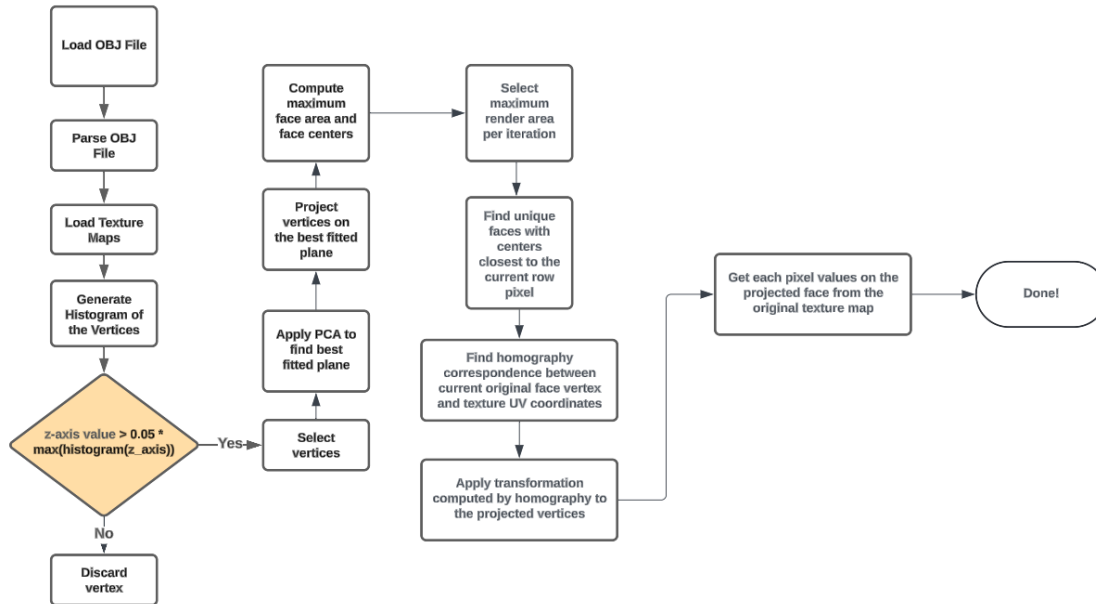


Figure 5.13 – Flowchart for Bridge Deck 2D Map Extraction from Point Clouds

Once a 2D deck map of bridges (either lane by lane or whole width of the bridge) is obtained, it is forwarded to a neural network for semantic segmentation and quantification of the inspector annotations.

5.4.2 Bridge Deck RGB Map Generation using Drone-Based RGB Images

Drone's RGB images can be used to develop a 2D map of whole bridge deck, then a deep learning model detects the deck delamination. **Figure 5.14** shows the proposed image stitching process using camera parameters, feature detection, and feature matching, to develop a 2D map of the entire bridge deck. In this method, RGB images are first rectified using camera calibration parameters (as discussed in **Sec. 5.3.4**). The rectified RGB images are then used to detect features within each image followed by the feature matching for corresponding points in the common regions of two overlapping images (as discussed in **Sec. 5.3.7**). Subsequently, a homography matrix (H_{RGB}) is calculated using these matched feature points (as discussed in **Sec. 5.3.5**). The process is performed on all images taken from the target bridge and then a 2D map of the bridge deck is formed by image warping and blending. Note that the 2D map of the bridge deck in **Fig. 5.14** was shown in a perspective view for clarity.

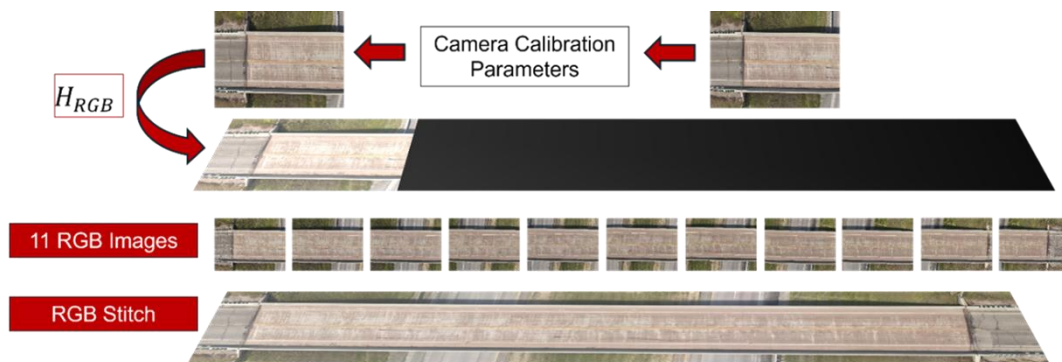


Figure 5.14 – Generation of 2D RGB Map for Entire Bridge Deck

5.5 AI Based Delamination Detection for Bridge Decks using 2D RGB Maps

When the 2D RGB map of a bridge deck is ready (using one of the two methods discussed in the previous sections), an AI model can be used to detect its delamination, which are paint marked by bridge inspectors. Note inspectors can use a chain-drag method or any other NDE methods to detect delamination in the field.

5.5.1 RGB Data Preparation for Model Training

The performance of any deep learning model depends on the quality and quantity of the data used for training and validation. Deep learning models generally perform better when trained on large datasets, as this improves their ability to generalize across diverse scenarios. Consequently, the training data must closely reflect real-world conditions to ensure accuracy and reliability. In this study, the neural networks were trained using the dataset described in **Ch. 4**. Specifically, 18 bridges were scanned using smartphone-based lidar, and more than 13,000 RGB images were collected from 50 bridges.

To better generalize features when the number of datapoints is limited (e.g., thousands instead of millions), several data augmentation techniques can be utilized to further diversify the dataset. In the present study, the training set was color augmented in the Hue, Saturation, and Value (HSV) color space to generate different variations of the paint colors when an inspector annotates a delamination. This image augmentation was applied randomly with a probability of

50%. Furthermore, different geometric augmentations such as scaling and rotation were applied. Scaling ranged from 0.5 to 1.0 of the training sample with a probability of 50%. A rotation of -45 to +45 degrees was also applied to increase the number of samples. **Figure 5.15** shows samples of raw and augmented data including their labels that were used in the model training.

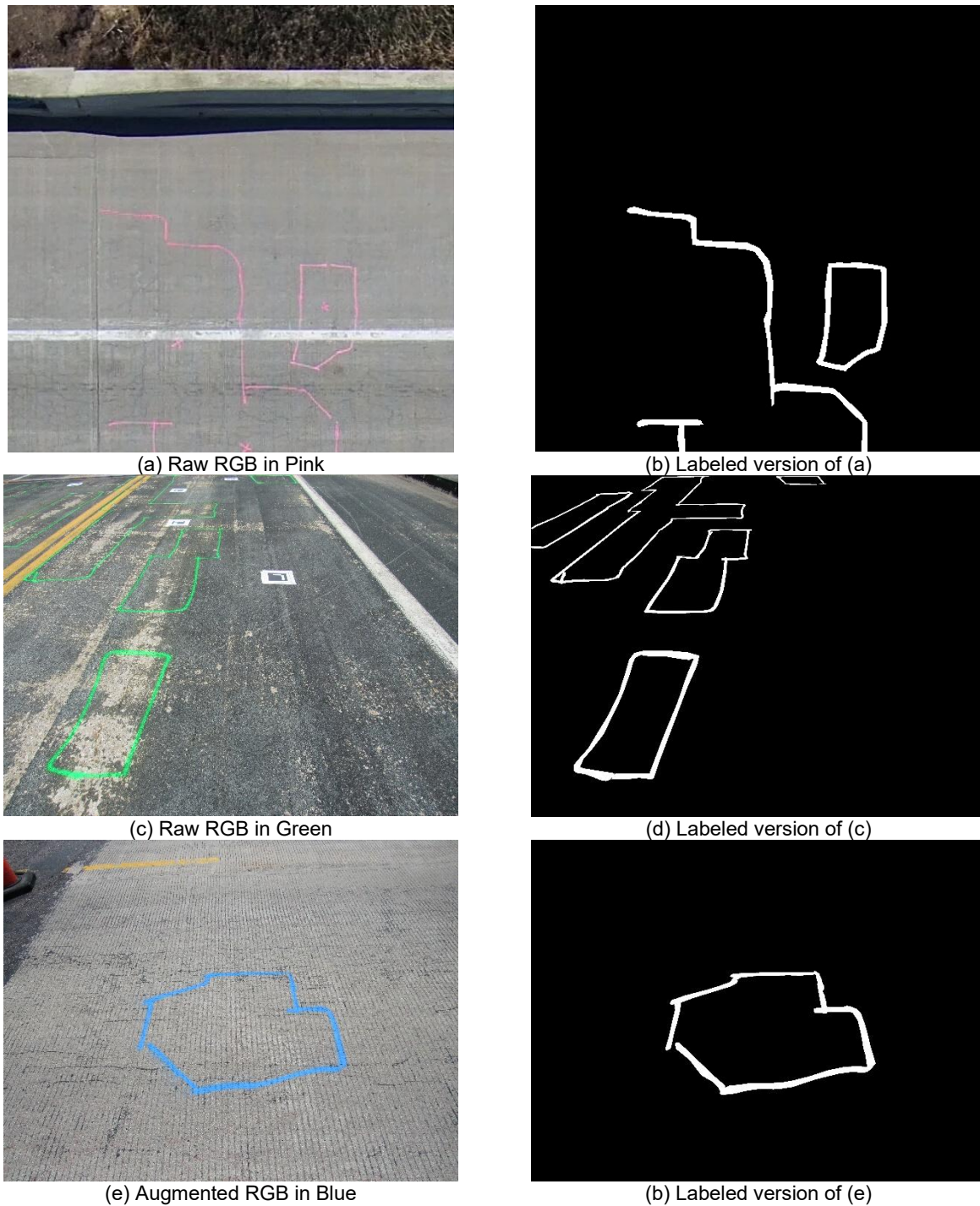


Figure 5.15 – Samples of Raw and Augmented Data and their Labels Used in Training

The final dataset was split into a training and testing set using the industry standard of 90% for training and 10% for testing. The training set was further split into 90% training and 10% validation sets. **Table 5.2** presents a summary of the inspector annotation RGB datasets used in the training, validation, and testing of the selected deep learning networks, which is discussed next. Note that the training set of 1,553 was increased to 70,000 after performing the above-mentioned augmentations.

Table 5.2 – Bridge Deck RGB Dataset Distribution Used in Neural Network

Set	Number of Samples
Training	1553
Validation	172
Testing	192

5.5.2 Selected Neural Networks for RGB Data Analysis

A deep learning network, U-Net (Ronneberger et al., 2015), was selected for semantic segmentation tasks of the present project to find the inspector annotations for delamination. U-Net (**Fig. 5.16**) is composed of one encoding path, a bottleneck channel, and a decoding path to classify pixels of images. This forms a U-shaped structure of the network. The encoding path involves a series of convolutional and max pooling layers, which progressively reduce the spatial dimensions of the input image while extracting hierarchical features. These layers act as an encoder, capturing context and high-level representations from the input image. The decoding path consists of upsampling and concatenation operations followed by convolutional layers. These layers gradually upsample the feature maps back to the original input resolution while incorporating high-resolution features from the contracting path through skip connections. The skip connections enable precise localization and refinement of segmentation boundaries by preserving fine-grained spatial information.

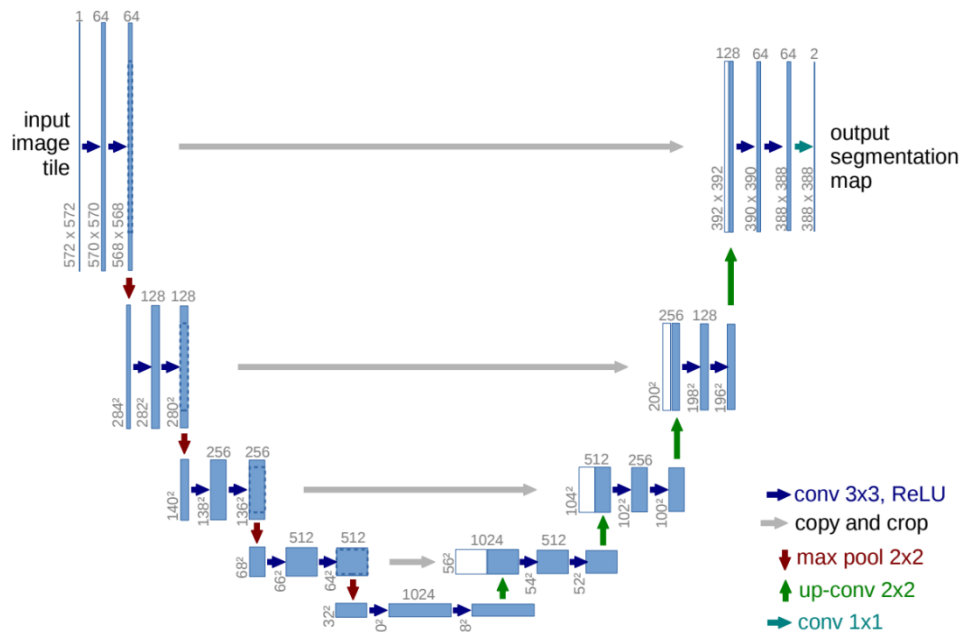


Figure 5.16 – U-Net Developed by Ronnenberger et al. (2015)

There are various versions of U-Net designed for different tasks and applications. In the present project, three variants of the U-Net model were explored for delamination detection: U-Net by Ronnenberger et al. (2015), Attention U-Net by Oktay et al. (2018), and Recurrent Residual U-Net by Alom et al. (2018).

The Attention U-Net extends the model by introducing attention gates as filters on the outputs of the skip connections as shown in **Fig. 5.17**. The structure of Attention U-Net retains the symmetric encoder-decoder design of the original U-Net model. In the encoder path, convolutional layers with ReLU activations and max-pooling operations are used to extract hierarchical features and downsample the input image. The decoder path employs upsampling and convolutional layers to reconstruct the image dimensions and refine the segmentation output. Attention gates are integrated into the skip connections between corresponding layers in the encoder and decoder paths. These gates learn to dynamically weigh the importance of different features, allowing the network to emphasize salient regions and ignore background noises.

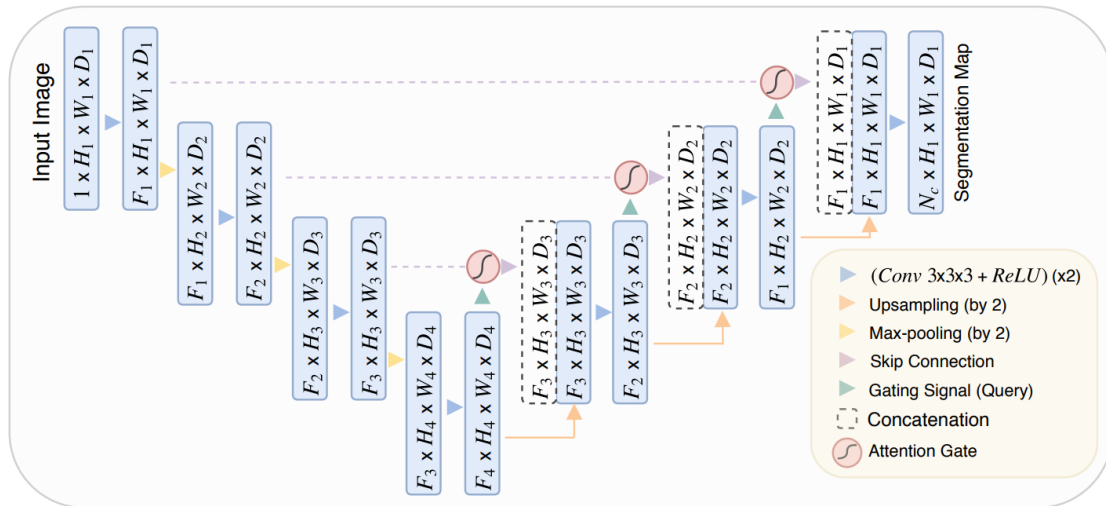


Figure 5.17 – Attention U-Net Framework Developed by Oktay et al. (2018)

The Recurrent Residual U-Net model extends U-Net by replacing the last convolution block of each layer with a recurrent convolution block with residual connection (**Fig. 5.18**). The model is better in capturing long-range spatial dependencies and contextual information because of the residual layers, which add temporal connections to the network. Recurrent feature processing allows the network to iteratively improve its comprehension of the input data. The residual connections offer shortcut paths that avoid one or more layers, which makes it easier to train deeper networks. This expedites convergence and aids in the solution of the vanishing gradient issue. Remaining connections are added to the encoder and decoder paths in Recurrent Residual U-Net, which enhances feature propagation and network stability.

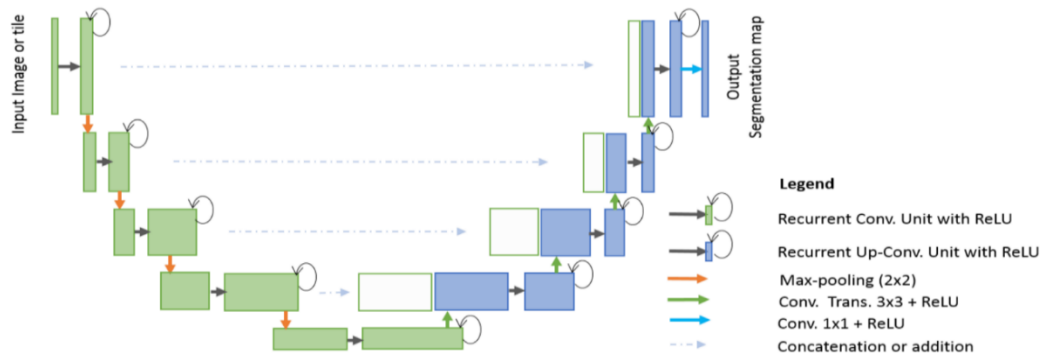


Figure 5.18 – Recurrent Residual U-Net Framework Developed by Alom et al. (2018)

5.5.3 Neural Network Model Training on RGB Dataset

For these three different models, different hyperparameters were used but with the same loss function, the Dice function. **Table 5.3** lists the best hyperparameters of the models during the training phase on the RGB dataset.

Table 5.3 – Hyperparameters of Evaluated Deep Learning Models During Training on RGB Dataset

Model	Batch Size	Learning Rate	Weight Decay	Momentum
U-Net	5	0.0001	1×10^{-8}	0.999
Attention U-Net	2	0.0005	1×10^{-8}	0.99
Recurrent Residual U-Net	4	0.0001	1×10^{-8}	0.99

The models are trained on the training set of the annotated dataset with each epoch having a validation stage on the validation dataset. Each validation stage aims to save the best-performing model parameters based on the validation set. **Figure 5.19** shows the trend of the validation score of the U-Net model. The best model had a score of 0.7375.

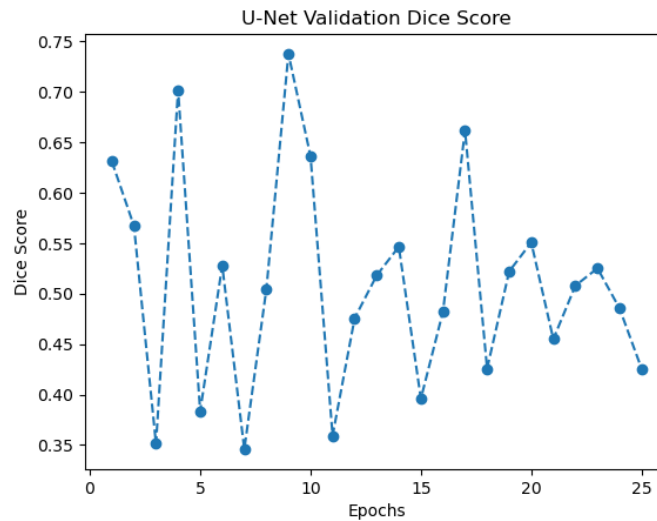


Figure 5.19 – U-Net Validation Score Trend during Training

Figure 5.20 shows the trend of the validation score of the Attention U-Net model in which the best model had a score of 0.6265.

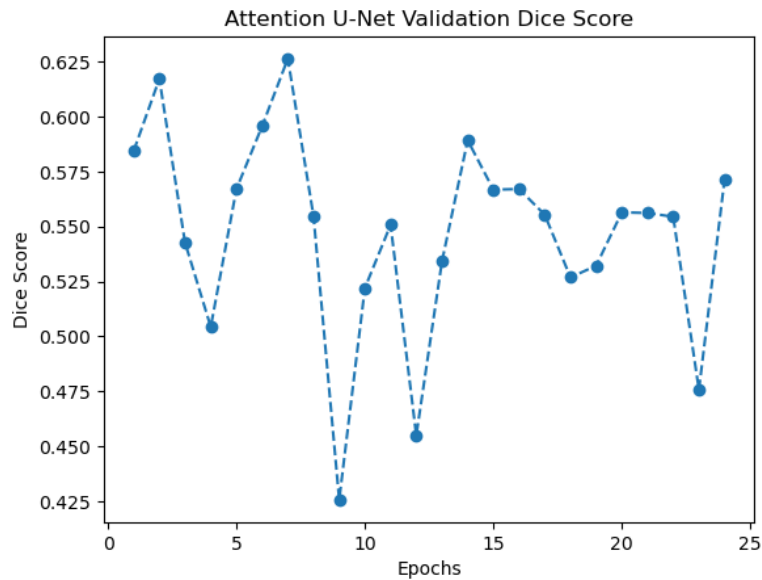


Figure 5.20 – Attention U-Net Validation Score Trend during Training

Figure 5.21 shows the trend of the validation score of the Recurrent Residual U-Net model in which the best model had a score of 0.6265.

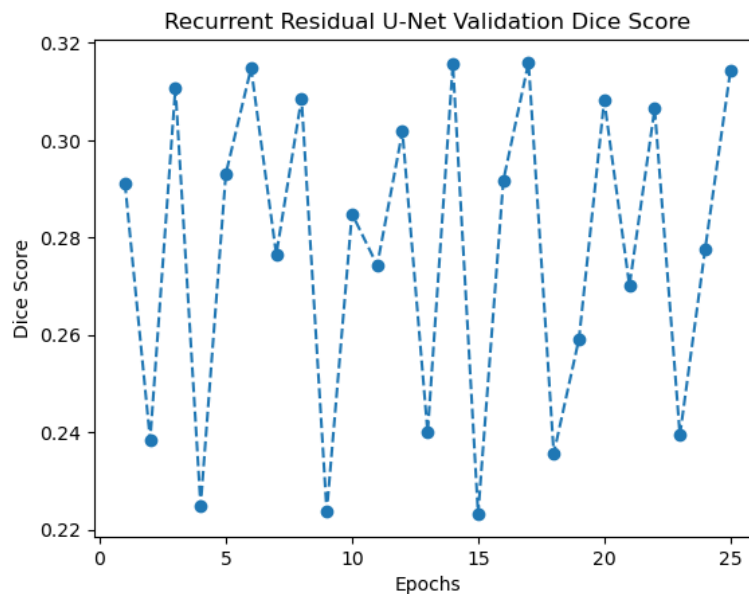


Figure 5.21 – Recurrent Residual U-Net Validation Score Trend during Training

The U-Net model achieved a validation score of 0.7375, indicating its effectiveness in segmenting the dataset. As shown in **Fig. 5.19**, the validation score of the U-Net model demonstrates a consistent improvement, culminating in its best score. The Attention U-Net and

the Recurrent Residual U-Net models both achieved a lower validation score. Despite the additional mechanisms designed to enhance segmentation performance in these models, such as attention mechanisms and recurrent residual connections, they did not surpass U-Net in terms of validation scores.

The results highlight that while advanced architectures such as the Attention U-Net and the Recurrent Residual U-Net incorporate sophisticated mechanisms aimed at improving segmentation, the classical U-Net model remains highly competitive and outperformed the other two models in this study. These findings suggest that the choice of model architecture should consider the specific dataset and task requirements, as more complex models do not necessarily guarantee a better performance. Further research and experimentation with different configurations and training strategies may help to improve the performance of these advanced models.

5.6 AI Based Delamination Detection on 2D RGB Maps from either Lidar or Drone Data

The research team initially developed two neural networks for analyzing RGB maps developed from either lidar data or drone imaging. Nevertheless, both were consolidated into a single model to simplify the process and the implementation in the inspection software.

5.7 AI Based Delamination Detection and Quantification in RGB Images

For the detection of delamination in RGB images, the three neural network models discussed in **Sec. 5.5.3** were further investigated at the local and global levels. The best model is used in the bridge inspection software. In this section, the delamination detection and quantification algorithms, and results of the model at local and global levels are discussed.

5.7.1 Detection of Delamination in RGB Images

The detection of delamination is performed on 2D RGB maps of bridge decks generated from the previous stage (**Sec. 5.4**). Since the generated RGB map is a very high-resolution image and can cover the whole length of the bridge, the image is split into small patches to reduce computational efforts by generating a finer segmentation of the target pixels. The trained U-Net models look for the pixels that fall within the annotation zone (delamination found and marked by inspectors). The annotation by the inspector ranges through different highlighting color spectrum. Usually, the region is marked with a water-based color spray. The model looks through each pixel of the input data to determine whether this is part of the annotation region or not. Once the entire bridge deck image is processed, the predicted masks are used to find each independent bounding region using a contour analysis. Each contour is organized hierarchically, enabling systematic processing to identify individual delaminated areas and calculate cumulative delamination across the entire RGB map. Each local deficiency is further analyzed using the proposed condition states by their area (**Ch. 3**). The following algorithm describes the process of computing the local area in pixels for each delamination:

Table 5.4 – Algorithm to Compute Delaminated Area

Input: Predicted Semantic Segmentation Map, *map*
Output: *local_damage_area*, *refined_contours*

Close any open polygon applying a 5x5 dilation kernel
processed_map := Apply Morphological Extension on *map*
hierarchy, *contours* := DetectContours (*processed_map*)

Refine Contours and Compute Local Damage Area
local_damage_area := []
foreach *contour* in *contours*:
 refined_contour := []
 foreach *point* in *contour*:
 refined_point := Find Coordinate of the Closest Pixel with Value == 1
 refined_contour := Append(*refined_contour*, *refined_point*)
 end
 refined_contours := Append(*refined_contours*, *refined_contour*)
 refined_area := PolygonArea(*refined_contours*)
 local_damage_area := Append(*local_damage_area*, *refined_area*)
end

5.7.2 Quantification of Delamination Condition States in RGB Images

Following the abovementioned method, the predicted delaminated areas are computed first in the pixel unit then are scaled to actual sizes (e.g., in² or mm²) using either the point cloud data in the lidar based method or a user-provide scale factor in the drone RGB based method. The quantification of the condition states for delamination on concrete decks was first presented in **Ch. 3**. A global flowchart of the inspection software with condition state estimation was discussed earlier in **Sec. 5.2**. However, **Fig. 5.22** shows a close-up of the flowchart detailing the steps taken to estimate the delamination condition states for bridge decks.

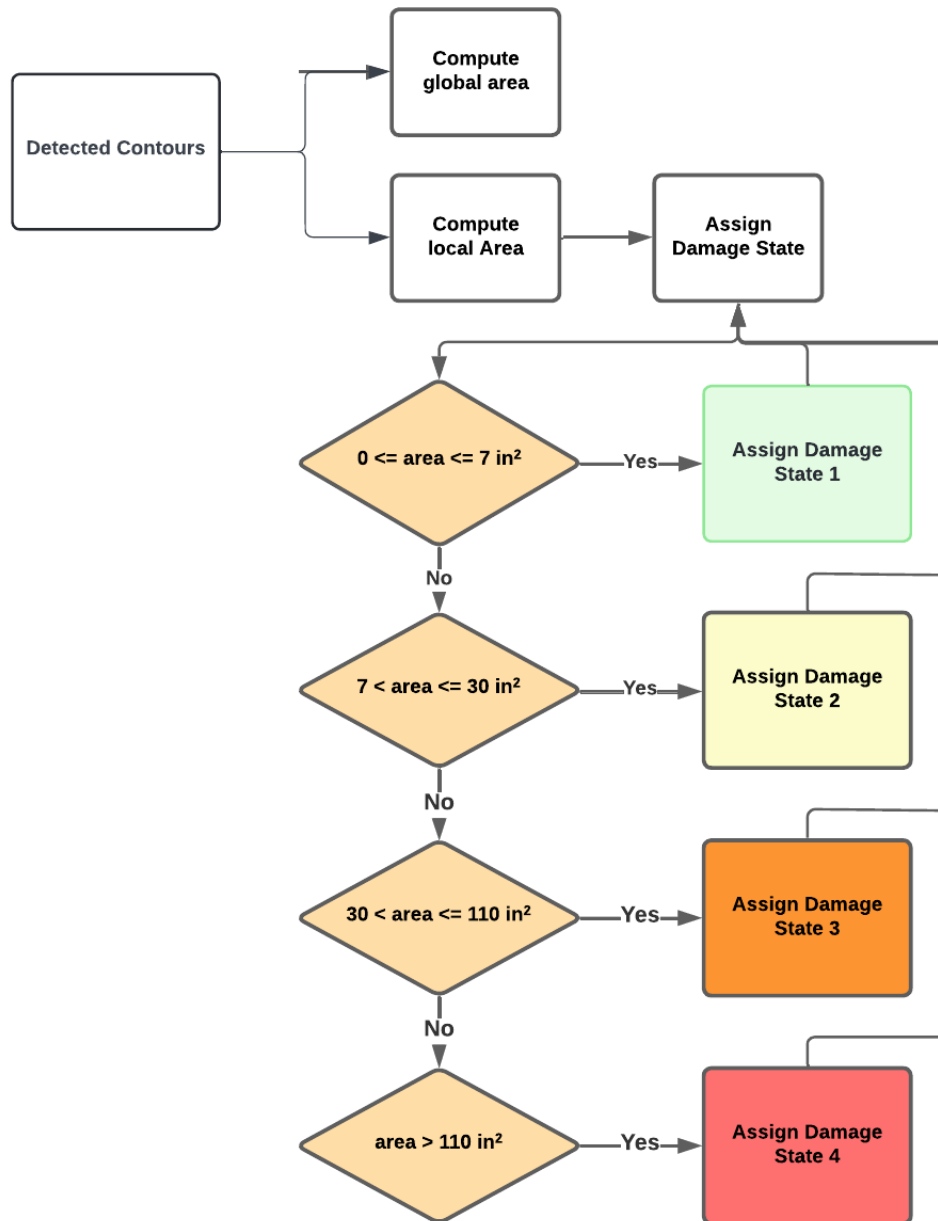


Figure 5.22 – Quantification of Condition States for Delamination

After computing the area of each delaminated region, the following algorithm was used to determine and label the condition state of each defect. The condition states are color coded according to AASHTO MBEI (2019) and then delaminated areas are overlaid on the top of the deck 2D map to form a complete damage map of the bridge deck.

Table 5.5 – Algorithm to Determine Delamination Condition State and Generate Damage Map

Input: *refined_contours, local_damage_area, area_in_pixels, area_in_metric*
Output: *damage_state_per_delam*

```
# Define conversion resolution from pixels to metric unit
resolution := area_in_inches / area_in_pixels

# Determine Damage States
damage_state_per_delam := []
foreach area in local_damage_area:
    unit_area := area * resolution
    # Damage State 1
    if unit_area ≤ 7:
        damage_state_per_delam := Append(damage_state_per_delam, 1)
    elif unit_area > 7 and unit_area ≤ 30:
        damage_state_per_delam := Append(damage_state_per_delam, 2)
    elif unit_area > 30 and unit_area ≤ 110:
        damage_state_per_delam := Append(damage_state_per_delam, 3)
    else
        damage_state_per_delam := Append(damage_state_per_delam, 4)
    endif
end
```

5.8 Results of AI-Based Delamination Detection in RGB Images

The performance of the models was evaluated on the test set of the dataset (individual images) as well as the complete 2D RGB maps of actual bridge decks, which were scanned with an iPhone and the maps were developed using the methods discussed in **Sec. 5.4**. Since each image of the testing set has one or a few delamination, such delamination can be assumed a “local” defect. However, a bridge deck might have many delaminated areas which must be evaluated. For ease of discussion, the deck-level delamination assessment is called “global” assessment. In other words, a local assessment is to determine the accuracy of the neural network models at the pixel level while the global assessment is to estimate the defects at the physical area level. The former was used during the training phase, and the latter was used during the testing phase of the network. The accuracy at both levels is discussed herein. Note that a pixel-to-pixel assessment was needed during the training of the models and to select the best model for delamination task. The section starts with the performance assessment of the neural networks on RGB maps obtained from the lidar method followed by a similar assessment on RGB maps obtained from the drone imaging.

5.8.1 Local Assessment of Neural Networks for Delamination Using iPhone Based Lidar Images

The local delamination was assessed on the patches of images with and without delamination. The Dice score can be used to evaluate such local performances. The Dice score is calculated using two other indices named “precision” and “recall”. The precision is the ratio between the detected true positive (TP) instances and the sum of true positives and false positive (FP) instances while the recall is the ratio of detected true positives and the sum of true positives and false negatives (FN) as:

$$Precision = \frac{TP}{TP + FP} \quad (5.19)$$

$$Recall = \frac{TP}{TP + FN} \quad (5.20)$$

In the present study, TP refers to the pixels in the ground truth image that are correctly identified as part of a delaminated area, which has been marked by inspectors during the inspection process. These are the pixels that both the inspectors and the model agree belong to a delaminated region, meaning the model has successfully detected the defect in the correct location. FP, on the other hand, represents the pixels that do not belong to a delaminated region in the ground truth image but have been incorrectly classified by the model as delamination. These are the pixels where the model mistakenly predicts delamination when none exists according to the inspector's annotation. FN is the number of pixels that are wrongly categorized as background or another object when they belong to the target object or region. The precision measures the accuracy of the positive predictions made by the model while the recall measures the ability of the model to find all the relevant pixels of the target object or region in the image. Subsequently, the Dice score (also known as the F-1 score) of the testing set is computed using the following equation:

$$Dice = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (5.21)$$

These local metrics were mostly assessed during the training of the neural models. **Table 5.6** presents a summary of the assessment for local (pixel-to-pixel) delamination. Furthermore, **Fig. 5.23** shows samples of the results at local views.

Table 5.6 – Local Assessment of Neural Networks for Bridge Deck Delamination Detection in RGB Images

Model	Test Set Performance (Dice Score at Pixel Level)
U-Net	73.75%
Attention U-Net	62.65%
Recurrent Residual U-Net	31.25%

CNNs work within a local area defined by a convolutional kernel to capture spatial hierarchies. This technique allows CNNs to learn features at different levels of abstraction, from low-level features (e.g., edges and textures) to high-level features (e.g., objects and patterns). While generating a segmentation mask for each pixel, the model determines the probability of that pixel belonging to a particular class based on its learned features and context from surrounding pixels. However, due to the complex nature of real-world images and the limitations of the model in perfectly capturing all nuances, it is challenging to achieve a complete pixel segmentation. Even the state-of-the-art models have limitations in fully capturing all possible variations of unseen data. Sometimes, CNN models may be overfitted during training thus they fail to perform well on new, unseen examples. There are also practical limitations on the size and complexity of the models that can be used, due to computational resources and the need for real-time performance in many applications.

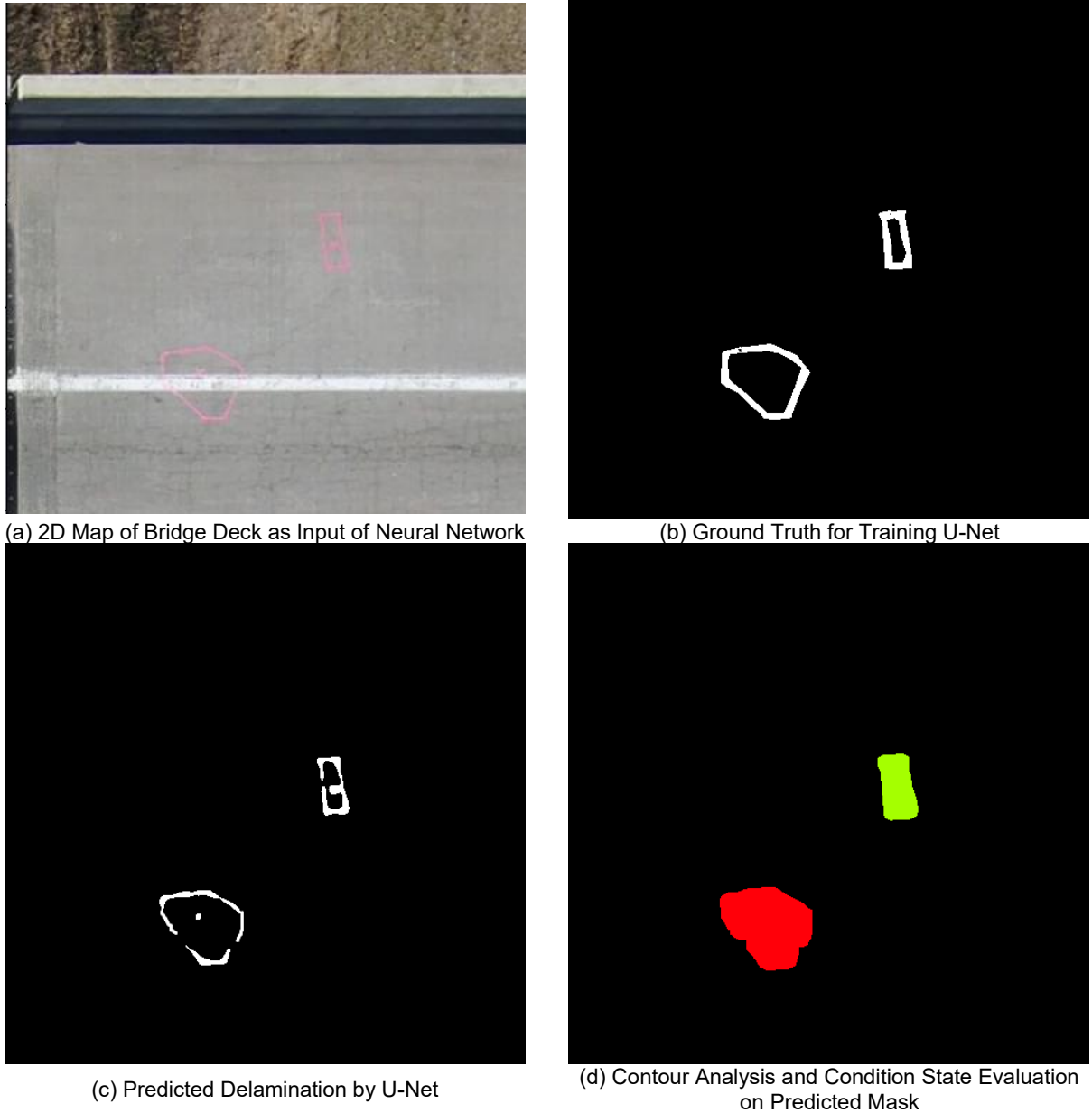


Figure 5.23 – Samples of Local Assessments of Neural Networks for Delamination

As discussed earlier, the performance of the neural networks in the present study was assessed at the local and global levels using different approaches. In the local assessment, a pixel-level assessment was done while in the global assessment the identified pixels were analyzed to enclose the damaged areas thus the estimated area was compared with the actual area, instead of finding just the pixels that represent the inspector annotations at the local level. It was found, as further discussed next, that the global assessment shows a higher accuracy than that of the local assessment. This is because finding the damaged area by connecting (enclosing) CNN detected pixels is less sensitive to the pixels that are missed (an area will be formed anyway), while a pixel-to-pixel comparison depends on a successful detection of every pixel. **Figure 5.23** better shows why the global assessments are more accurate than the local estimates. **Figure 5.23a**

shows a slice of the bridge deck that has two delamination annotated by the inspector. **Figure 5.23b** shows the ground truth mask for the annotation. **Figure 5.23c** shows the pixel-wise prediction by the model where the model did not detect all the pixels which were part of the delaminated area. After performing a contour analysis, the boundary of the damaged region was found followed by a condition state analysis as shown in **Fig. 5.23d**. It is clear that when comparing the estimated delaminated area (**Fig. 5.23d**) with the actual delaminated area (**Fig. 5.23b**), the number of missed pixels was not important (as the contour function enclosed the gaps and computed the area of the enclosed polygon) thus a global assessment is generally more accurate than a pixel-to-pixel assessment (compare **Fig. 5.23c** with **Fig. 5.23b**). Note that the accuracy presented in **Table 5.6** was at the pixel level not the area level, which is discussed next.

5.8.2 Global Assessment of Neural Networks for Delamination Using iPhone Based Lidar Images

A delamination analysis at the global scale (e.g., area-to-area at the full length of the bridge) provides a complete assessment of the entire processes and methods of autonomous delamination detection in real applications. **Figure 5.24** shows one sample of delamination assessment for a bridge on its full length, which was 248 ft (75.4 m).

Figure 5.24a shows the input image, a 2D map of the bridge deck including the marked delamination by the inspector (the enclosed areas with green paint), to the neural network. This input image is then processed to extract patches with a size of 512-by-512 pixels with a stride of 256. This makes each patch to have a 50% overlap with the adjacent ones.

Figures 5.24b-d show the predicted pixels by the three neural networks as the delamination boundaries (finding the green markups automatically). The findings are then used to compute the delamination condition state for individual zone using a contour analysis on the post-processed prediction map. The predicted delaminated pixels are then connected using morphological transformations and passed through a contour analysis. After the completion of the contour analysis, which was done to enclose each delamination and to obtain its shape, each of the delaminated areas is then color coded based on the severity of the delamination (**Sec. 3.3**). The scale information was retrieved from the mesh (point cloud) data.

A similar evaluation was performed on 18 different bridges to evaluate the framework. **Table 5.7** presents a summary of the findings at the global level for these 18 bridges. The accuracy of the U-Net model was extremely high (with only 1% error) for these real-world applications.

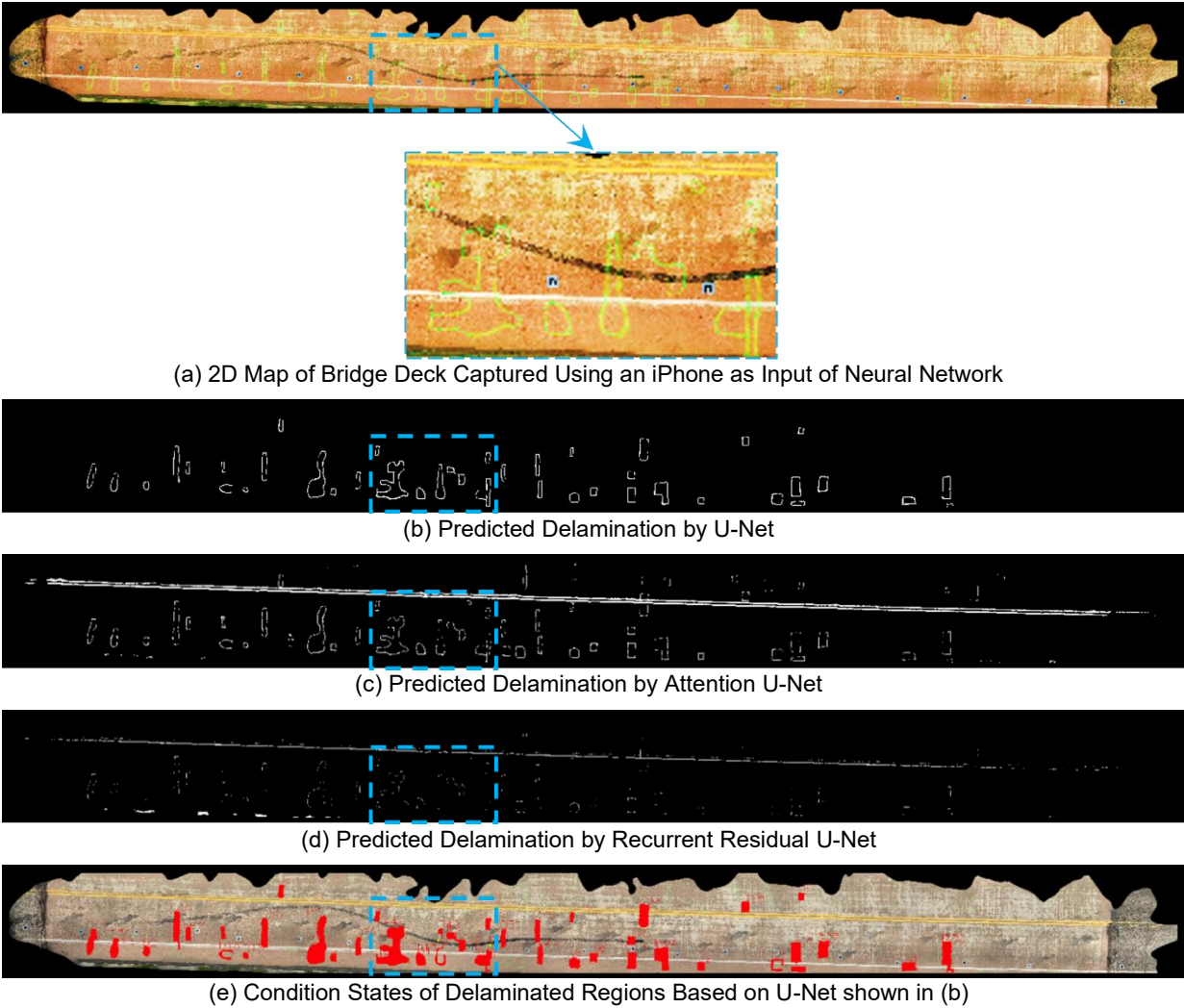
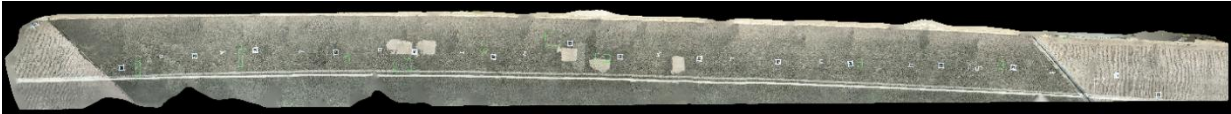


Figure 5.24 – Delamination Assessment of a 75.4-m (248-ft) Bridge Using Lidar Imaging

Table 5.7 – Global Assessment of Neural Networks for Delamination Detection in RGB Maps of 18 Bridge Decks

Model	Deck Map Performance (Area)
U-Net	99.5%
Attention U-Net	84.5%
Recurrent Residual U-Net	< 5%

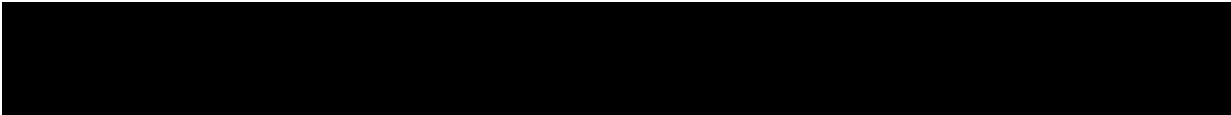
Figures 5.25 and 5.26 show the results of similar area-level assessments for two more bridge decks, one curved and another skewed. It can be seen that in all these cases, U-Net showed a better performance compared with the other two models.



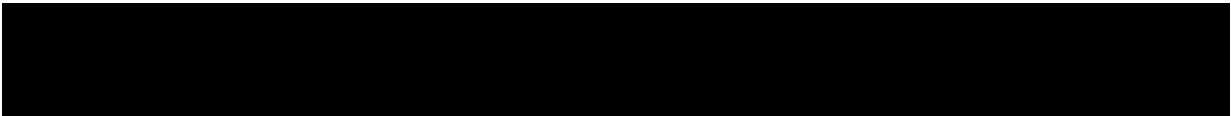
(a) 2D Map of Bridge Deck Captured Using an iPhone as Input of Neural Network



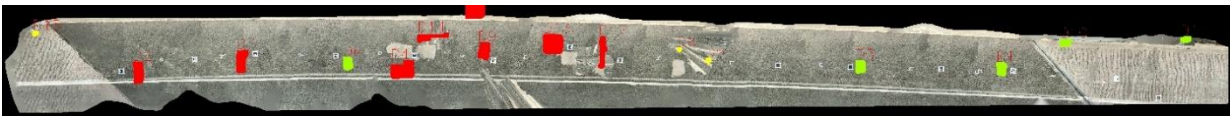
(b) Predicted Delamination by U-Net



(c) Predicted Delamination by Attention U-Net



(d) Predicted Delamination by Recurrent Residual U-Net

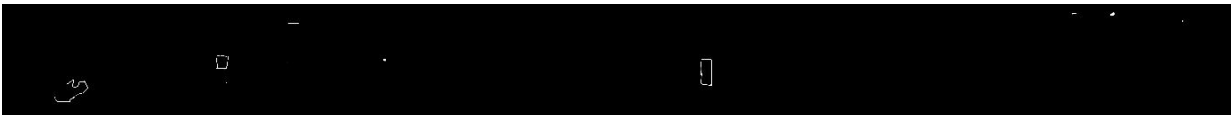


(e) Condition States of Delaminated Regions Based on U-Net shown in (b)

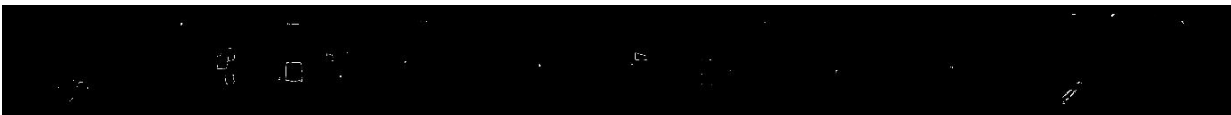
Figure 5.25 – Delamination Assessment of a 52.7-m (173-ft) Curved Bridge Using Lidar Imaging



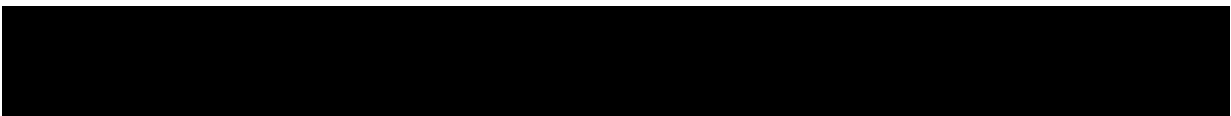
(a) 2D Map of Bridge Deck Captured Using an iPhone as Input of Neural Network



(b) Predicted Delamination by U-Net



(c) Predicted Delamination by Attention U-Net



(d) Predicted Delamination by Recurrent Residual U-Net



(e) Condition States of Delaminated Regions Based on U-Net shown in (b)

Figure 5.26 – Delamination Assessment of a 52.4-m (172-ft) Skewed Bridge Using Lidar Imaging

Table 5.8 presents a summary of the local (pixel to pixel) and global (area to area) evaluations. The U-Net and Attention U-Net models performed well on the test set while the Recurrent Residual U-Net showed an inferior performance on the test and deck map sets. Overall, U-Net performed better than the other two models both on the local and global scales and is recommended for field applications.

Table 5.8 – Performance of Neural Networks in Detection of Bridge Deck Delamination in RGB Images

Model	Test Set (Local) Performance (Dice Score at Pixel Level)	Deck Map (Global) Performance (Area Level) for 18 Bridges
U-Net	73.75%	99.5%
Attention U-Net	62.65%	84.5%
Recurrent Residual U-Net	31.25%	< 5%

5.8.3 Global Assessment of Neural Networks for Delamination Using Drone Based RGB Images

A high elevation image of bridge decks can cover the entire bridge deck; however, the resolution of such image is usually low limiting its application in neural networks. By lowering the flying altitude of the drone, a high-resolution image can be captured; however, it may not cover the full length of the bridge due to the field of view of drones' cameras. Overlapping images can be collected and those images can be stitched together using homography matrices estimated based on the image feature detection and matching in successive frames, which was discussed earlier in **Sec. 5.4.2**. **Figure 27a** shows an example of a stitched image using 50 individual images of a 293 ft (89.3 m) long bridge deck. In this image, the delaminated regions are marked with circles painted on the bridge deck. The aim is to identify these green circles in the stitched image using the best trained neural network, U-Net. The neural network successfully detected the paint markings in the image (**Fig. 27b**). The accuracy of the neural network in detecting the defected areas was 96% compared with that of the actual damage. **Figure 27c** provides an overlaid delamination map on the bridge deck RGB map.

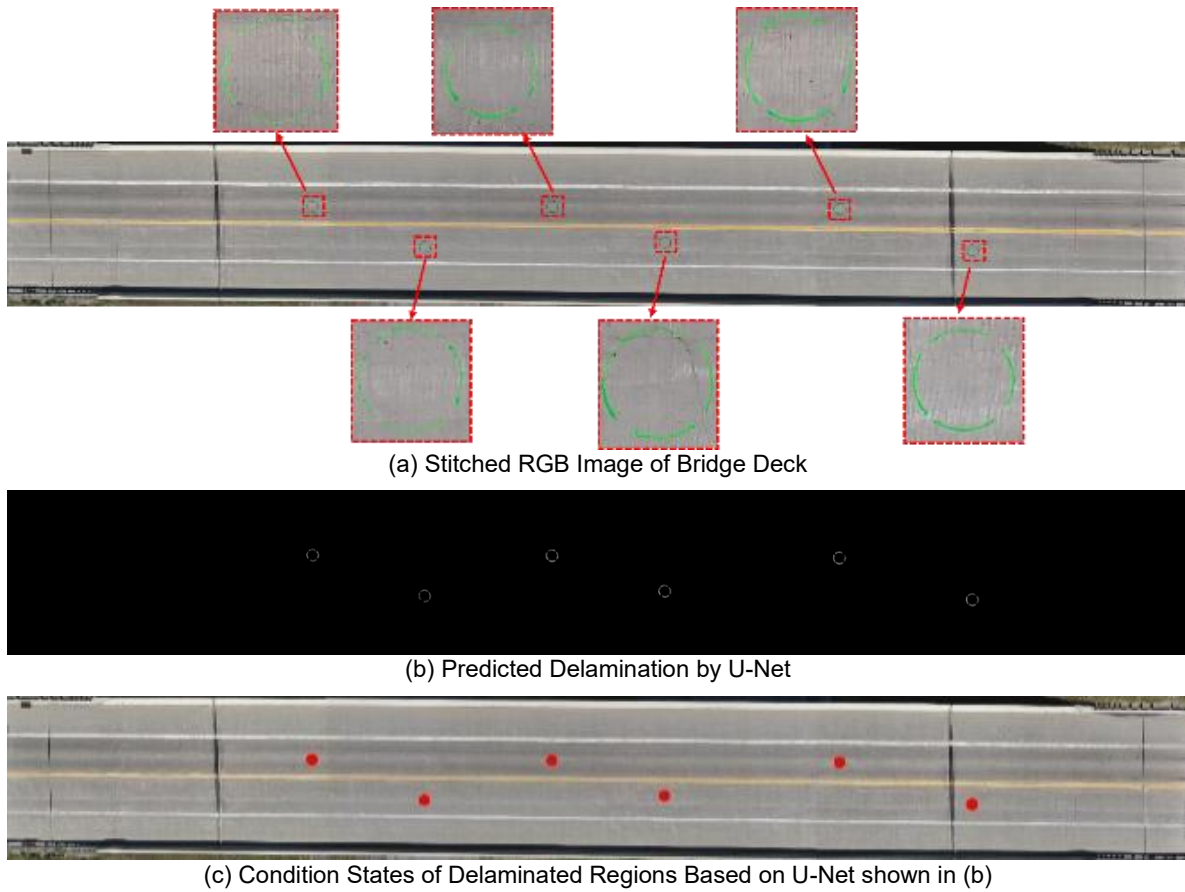


Figure 5.27 – Delamination Assessment of an 89.3-m (293-ft) Bridge Using Drone RGB Imaging

Overall, the study confirms that a 2D RGB map of bridge decks can be constructed using either lidar or drone imaging. Furthermore, the trained U-Net model of the present study is able to detect the inspector markups from RGB images and maps with very high accuracy at the area level of bridge decks (more than 95%).

CHAPTER 6 – THERMAL BASED AI DELAMINATION DETECTION

6.1 Introduction

Thermal images can show internal defects of concrete such as delamination (e.g., as hot spots if data is collected in daytime) compared with the undamaged regions. The current state-of-the-art thermal image collection and processing lacks systematic and automated methods for bridges. In the present study, first a novel approach to develop a thermal map of the entire bridge deck is developed. The new thermal map includes the actual temperature of bridge decks instead of thermal intensity that was used in past studies. Subsequently, neural networks were trained to detect delaminated regions using deck thermal maps.

This chapter presents a methodology to detect concrete delamination from thermal maps of bridge decks. The components of such methodology, such as thermal map generation for decks, thermal data normalization, neural network training and accuracy, etc., are discussed in the following sections of the chapter. The best neural network and normalization method were used to generate delamination maps for concrete bridge decks.

6.2 Proposed Methodology to Detect Bridge Deck Delamination in Thermal Images

Figure 6.1 shows the proposed methodology to automatically generate a delamination map for concrete bridge decks using a fused RGB-thermal imaging and computer vision. The steps include collection of images using drones each equipped with a pair of thermal and RGB cameras, which are calibrated and correlated. Subsequently, RGB images are stitched to develop a two-dimensional (2D) map of the bridge deck. Then, thermal data are superimposed over the RGB data using homography mapping. Subsequently, the fused data is processed to obtain the thermal map of the whole bridge deck. The final stage is to analysis the thermal map using a neural network to automatically detect delaminated regions. As a result of this process, a delamination map for the entire bridge deck is generated. The following sections discuss these key steps to obtain thermal maps of bridge decks and to detect delamination using AI.

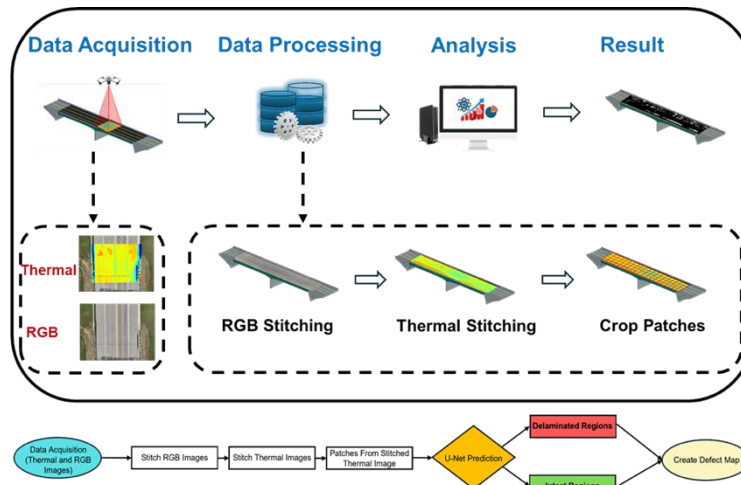


Figure 6.1 – Workflow for Delamination Detection in Concrete Bridge Decks Using Thermal Images

6.3 Proposed Methodology to Develop Bridge Deck Thermal Map

Figure 6.2 shows the complete image stitching process using camera parameters, feature detection and matching, and homography (all discussed earlier in **Sec. 5.3**) to develop a thermal map of the entire bridge deck. As an example, and to show the process, the figure shows the thermal map generation using 11 pairs of RGB and thermal images. RGB images were rectified using the camera calibration parameters. The rectified RGB images were used to detect image features followed by feature matchings for corresponding points in the overlapping regions. Homography matrices were estimated to stitch RGB images. Note that the complete RGB image stitching method was discussed in **Sec. 5.4.2**. Followed by the RGB map generation, thermal images were also rectified using thermal camera calibration parameters. Another homography matrix was estimated to warp thermal images on their corresponding RGB images. These warped thermal images were then stitched together using homography matrices already calculated for RGB image stitching. Average temperature values were used in the overlapping regions of warped thermal images. **Figure 6.3** shows the final stitched RGB and thermal maps for the entire deck of this sample bridge.

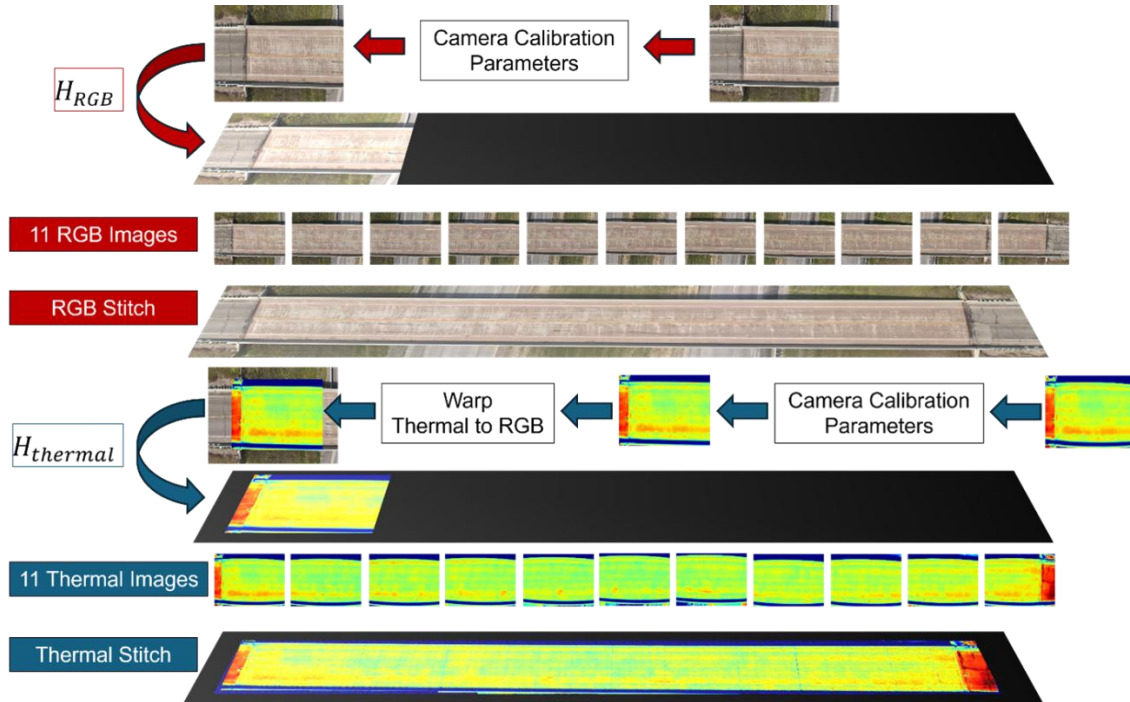


Figure 6.2 – Generation of 2D Thermal Maps for Entire Bridge Decks

Note that the field of view for thermal cameras is usually narrower than that of RGB cameras. Therefore, thermal images should cover the area of interest as a minimum. Camera calibration parameters also play a vital role in image stitching as it rectifies the errors in the image, and the lines get straightened at the edges. Without camera calibration, the straight lines appear as curves causing mismatch in the image stitching process.

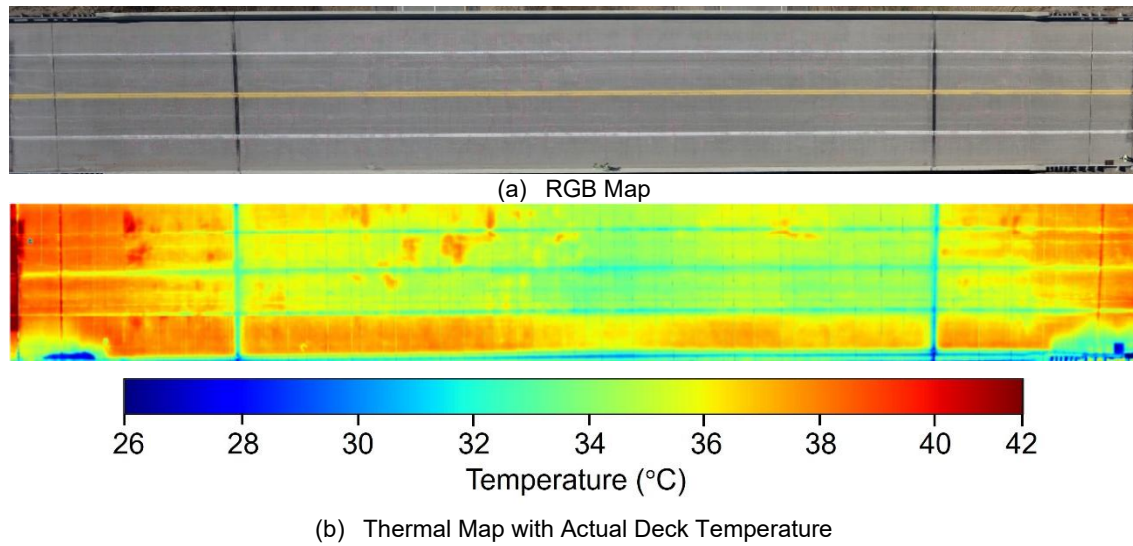


Figure 6.3 – Final Stitched RGB and Thermal Maps for Bridge Shown in Fig. 6.2

During data collection phase, two orientations of drone's camera with respect to the bridge longitudinal direction were exercised as shown in **Fig. 6.4**. In Case-A, the wider view of the drone's camera was in the bridge transverse direction and in Case-B, the wider view of the drone's camera was in the bridge longitudinal direction. These different orientations result in covering different areas of interest. In Case-A, a greater number of images are required to cover the entire length of the bridge. The proposed method for creating RGB and thermal maps of entire bridge length works well in both scenarios. However, the current commercial drones are equipped with thermal cameras that are much narrower in their fields of view compared with those of RGB cameras. The study recommends using the field of view of thermal cameras as the baselines of RGB-thermal data collection.

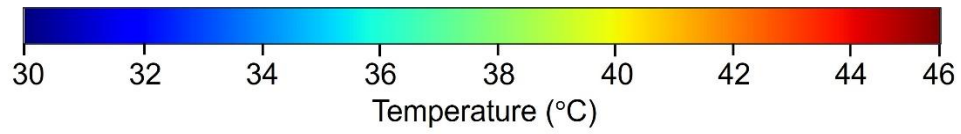
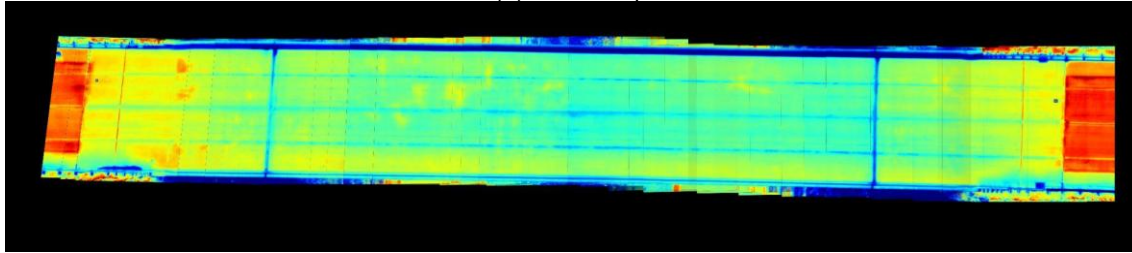


Figure 6.4 – Effects of Drones' Field of View on Images

The proposed method to develop thermal maps for bridge decks using the fused RGB-thermal imaging was successfully exercised on several other bridges, some examples are shown in **Fig. 6.5 to 6.11**.



(a) RGB Map



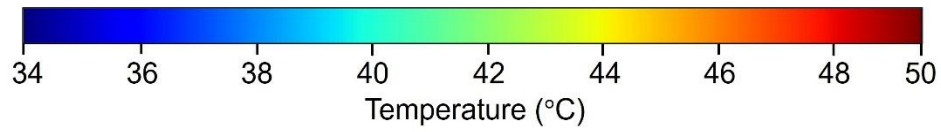
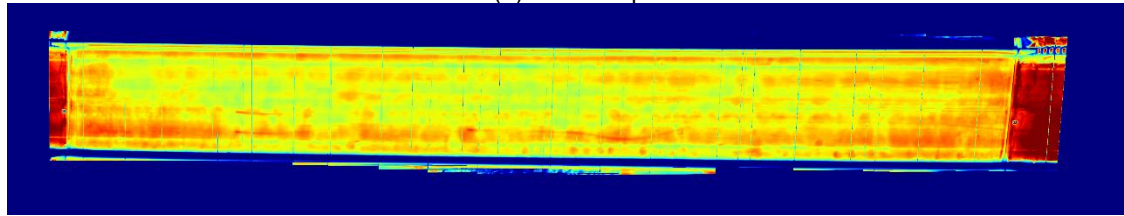
(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
15215030	89.3 (293)	13.4 (44)	2L-2S	3.65 (12)	3 (10)	16.5 (54)	36	25-50	Case-A	5453×1091

Figure 6.5 – Stitched Deck RGB and Thermal Maps for Bridge 15215030



(a) RGB Map



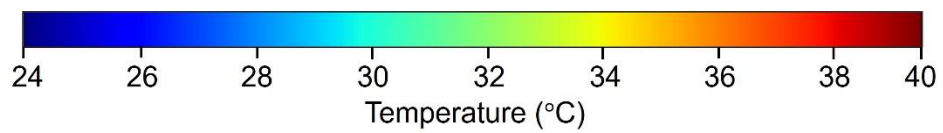
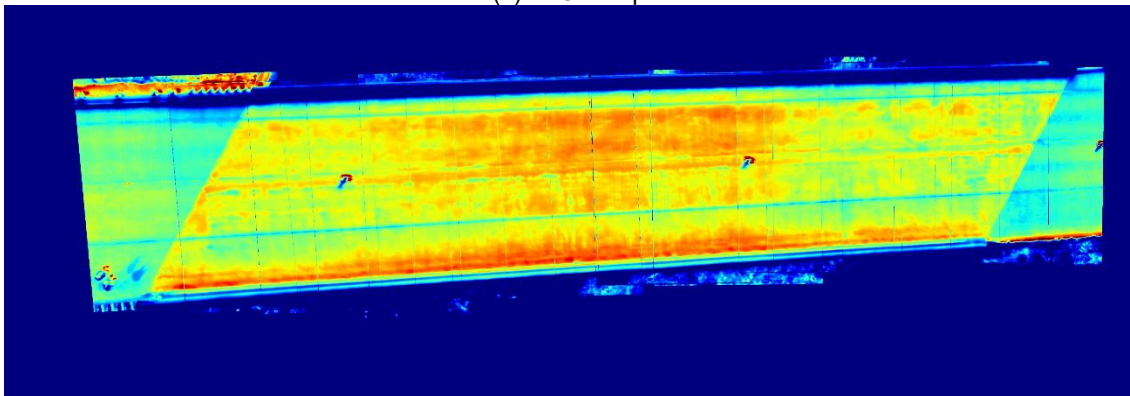
(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
06185230	7.4 (253.9)	9.1 (29.9)	2L	3.65 (12)	0.9 (2.95)	15.5 (51)	20	50-60	Case-B	4239×798

Figure 6.6 – Stitched Deck RGB and Thermal Maps for Bridge 06185230



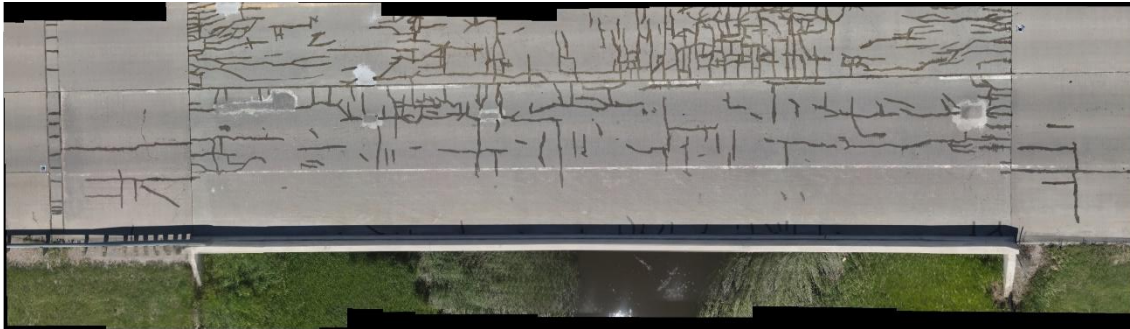
(a) RGB Map



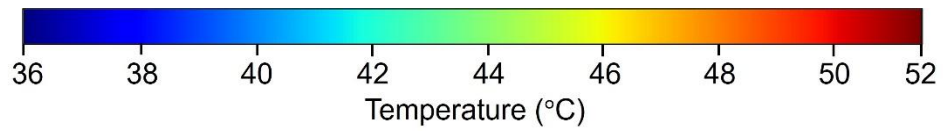
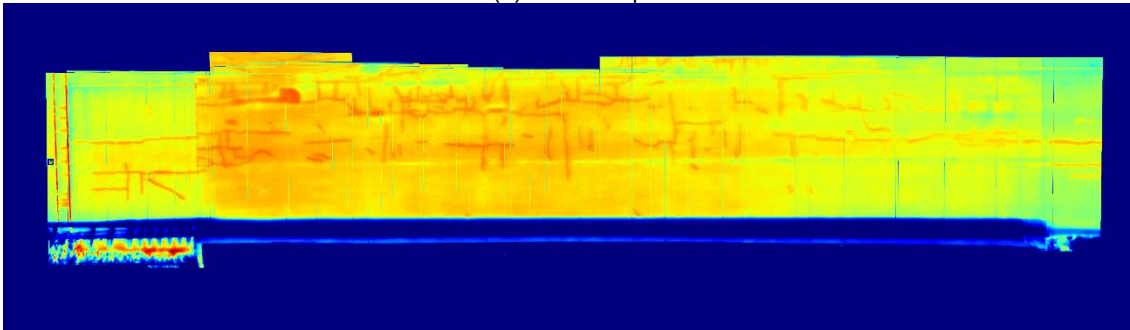
(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
03239216	52.2 (171.3)	11.6 (38.1)	2L-S	3.65 (12)	3 (10)	15.4 (50.5)	15	25-50	Case-A	3534×1219

Figure 6.7 – Stitched Deck RGB and Thermal Maps for Bridge 03239216



(a) RGB Map



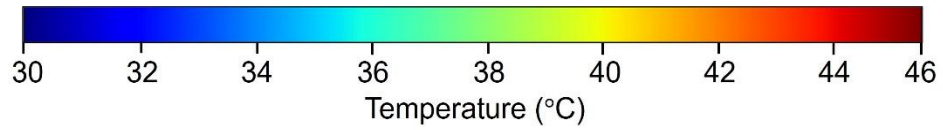
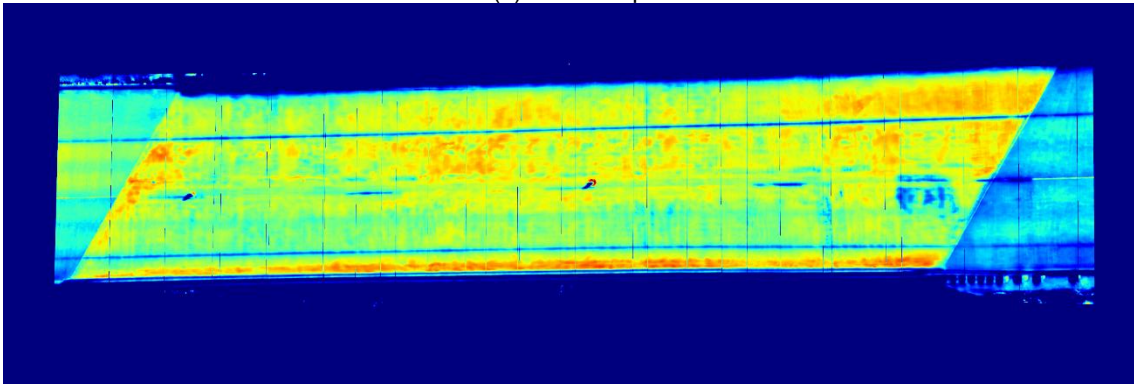
(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
06158158	36.3 (119.1)	11.6 (38.1)	2L-2S	3.65 (12)	3 (10)	9.4 (30.8)	27	60-75	Case-A	3870×1120

Figure 6.8 – Stitched Deck RGB and Thermal Maps for Bridge 06158158



(a) RGB Map



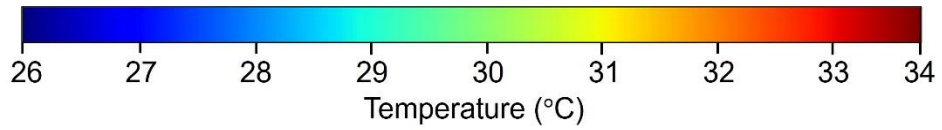
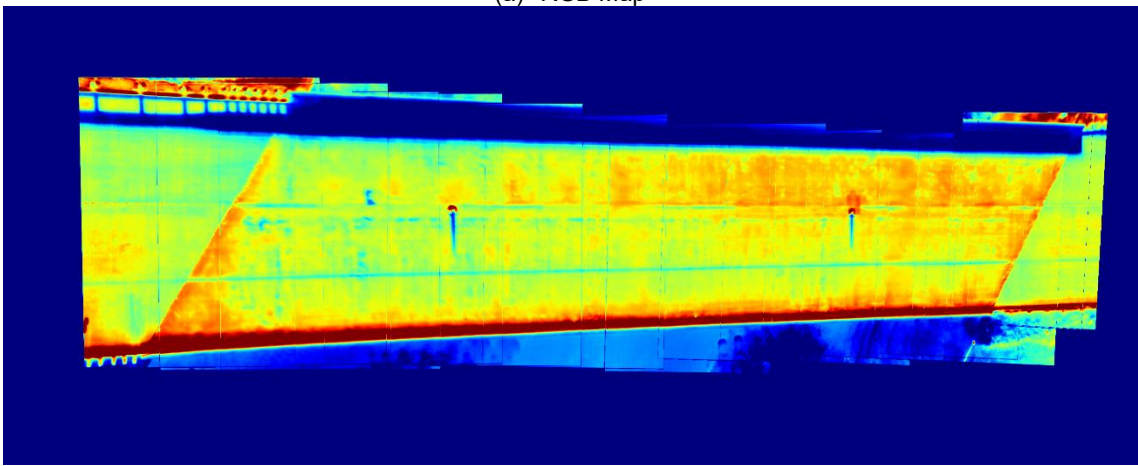
(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
03240216	52.2 (171.3)	11.6 (38.1)	2L-1S	3.65 (12)	3 (10)	15.3 (50.3)	17	25-50	Case-A	3233×1085

Figure 6.9 – Stitched Deck RGB and Thermal Maps for Bridge 03240216



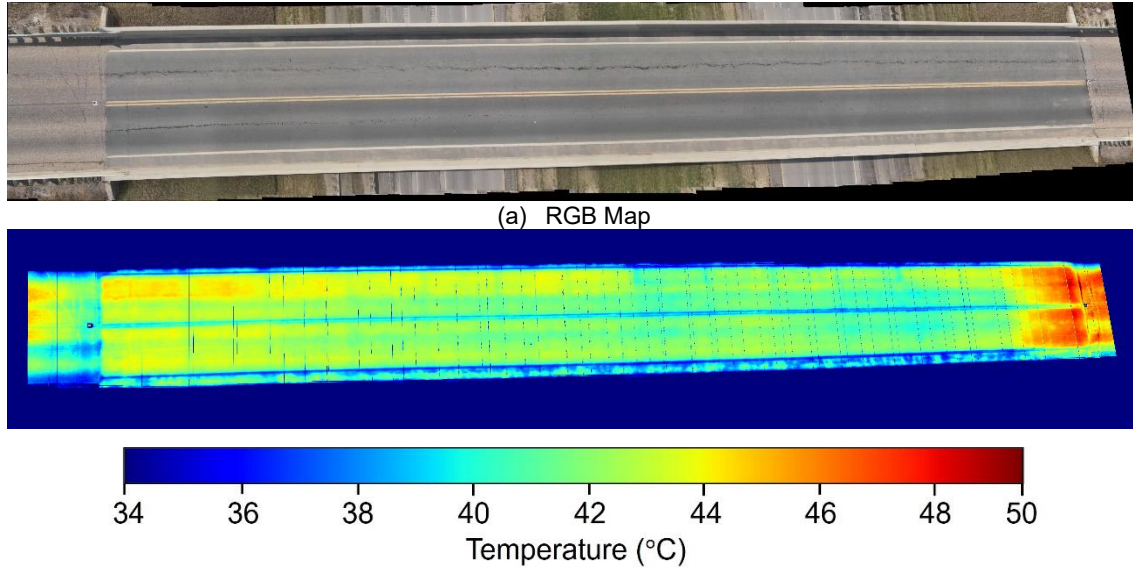
(a) RGB Map



(b) Thermal Map

Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
03239257	52.2 (171.3)	11.6 (38.1)	2L-1S	3.65 (12)	3 (10)	15.4 (50.5)	17	25-50	Case-A	2678×1082

Figure 6.10 – Stitched Deck RGB and Thermal Maps for Bridge 03239257



Bridge ID	Length m (ft)	Width m (ft)	No. of Lanes (L) and Shoulders (S)	Lane Width m (ft)	Shoulder Width m (ft)	Height of Drone m (ft)	No. of Images	Image Overlap (%)	Image Orientation Relative to Bridge	Output (pixels)
06185190	77.4 (253.9)	8.5 (27.9)	2L	3.65 (12)	0.6 (1.95)	9.8 (32)	36	25-50	Case-A	5856×1026

Figure 6.11 – Stitched Deck RGB and Thermal Maps for Bridge 06185190

6.4 Perspective View of Stitched Images

The perspective view of stitched images can be corrected to be in the plan view of the deck for the ease of analysis. This can be done by selecting four points of the bridge deck (e.g., at bridge ends). Based on these points, perspective transformation matrix is estimated. This matrix is used to map all the points in the distorted perspective view to their corresponding locations in the corrected (or desired) view (the plan view) of the stitched image:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = T \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (6.1)$$

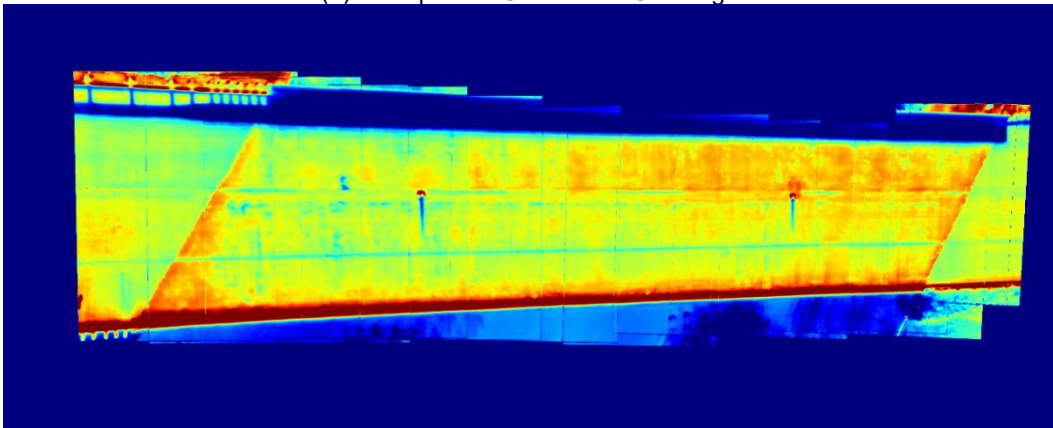
where, x and y are the coordinates of a point in the source image (distorted perspective view), and x' and y' are the coordinates of the corresponding point in the destination image (corrected plan view). The perspective transformation matrix, T , is a 3×3 matrix. **Figure 6.12** presents a sample of a corrected perspective for a bridge.



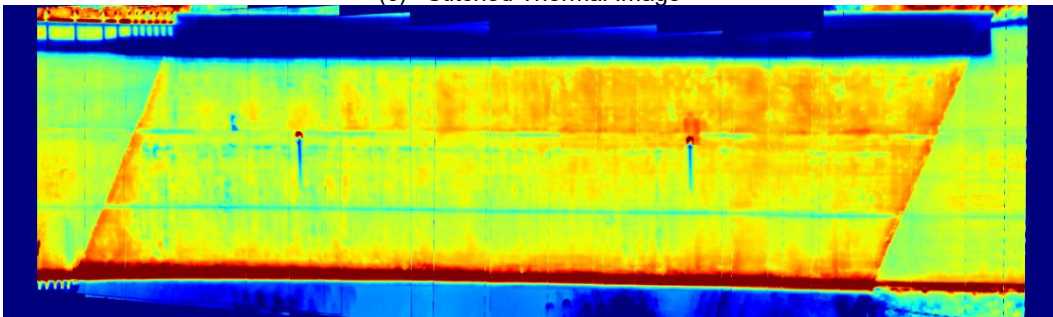
(a) Stitched RGB Image



(b) Perspective Corrected RGB Image



(c) Stitched Thermal Image



(d) Perspective Corrected Thermal Image

Figure 6.12 – Perspective Corrected RGB and Thermal Maps of a Bridge Deck

6.5 AI Based Delamination Detection for Bridge Decks using 2D Thermal Maps

When the 2D thermal map of a bridge deck is ready, an AI model can be used to detect its delamination. Note that in the thermal method, inspector annotations are not needed, which can significantly expedite field inspections.

6.5.1 Selected Neural Networks for Thermal Data Analysis

For thermal images, U-Net, a deep convolutional neural network (DCNN), was used as the base model to predict delamination in thermal images. As discussed in **Ch. 5**, U-Net was developed by Ronneberger et al. (2015) to perform semantic segmentation in the medical field. U-Net has a U-shaped encoder-decoder neural network architecture (e.g., **Fig. 6.13** for thermal images). In the encoder part, double convolutional blocks were used along with max pooling layers. Rectified linear unit (ReLU) activation function was used between the convolutional layers. The kernel size was 3×3 for each convolutional layer. In the decoder part, transposed convolution layers with a kernel size of 2×2 were added. **Figure 6.13** shows the U-Net architecture used for thermal image dataset in the study. Due to the model limitations, the input thermal images were resized from 640×512 pixels to 352×352 pixels. Furthermore, since the dataset was unbalanced, large pixels were related to the solid (undamaged) regions as compared with the delaminated regions, and the Dice loss function was used, which performs better for unbalanced dataset (Li et al., 2019). No pre-trained model was used.

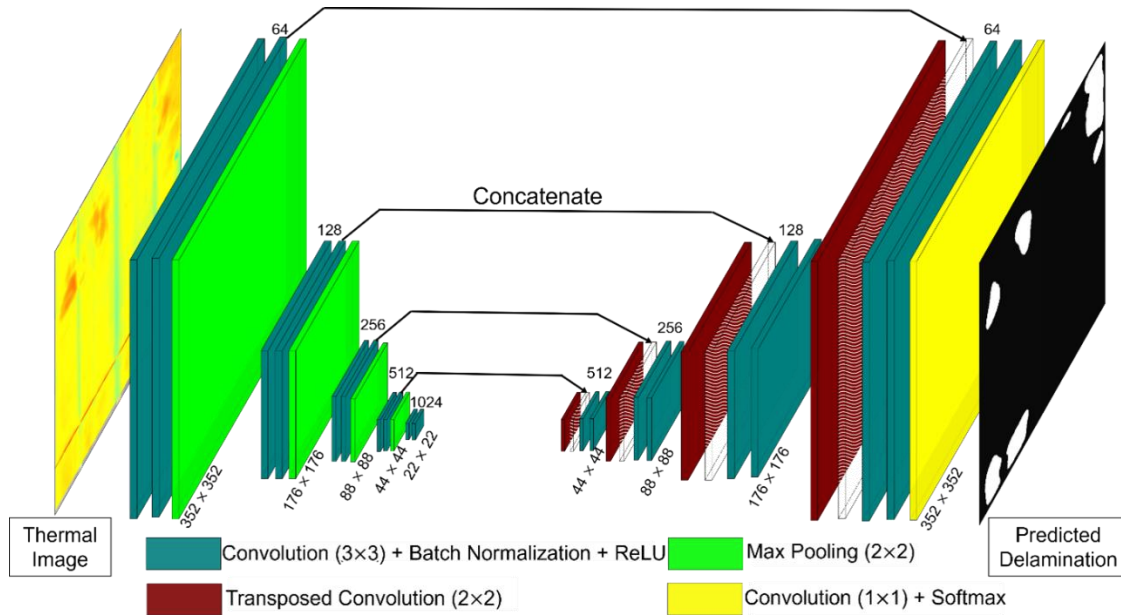


Figure 6.13 – U-Net Architecture for Thermal Images

In addition to U-Net, five variants of the U-Net architecture (Attention U-Net, Residual U-Net, Transformer ResU-Net, Recurrent Residual U-Net, and ResUNet++) were selected for performance comparison against the base U-Net model.

6.5.1.1 Attention U-Net

Attention U-Net (**Fig. 6.14**) incorporates attention gate mechanism inside its architecture which helps the network to focus on pertinent regions of the image which can potentially improve the performance of the network (Oktay et al., 2018). The attention gate (**Fig 6.15**) takes the input from the previous decoder layer and a skip connection from the encoder layer which has high spatial resolution. The attention gate reduces the dimension of the skip connection to match the dimension of the input from the lower decoder layer to add both the inputs. This addition increases the value of aligned weights thereby putting more attention to those regions relative to the other parts of the input. Then ReLU activation layer keeps the range of values between 0 to infinity. After 1×1 convolution, a sigmoid layer is used to bring the values in the range of 0 to 1. Resampler matches the output to the size of skip connection so that point wise multiplication can be done. This complete procedure helps the network to pay more attention to important regions in the input image.

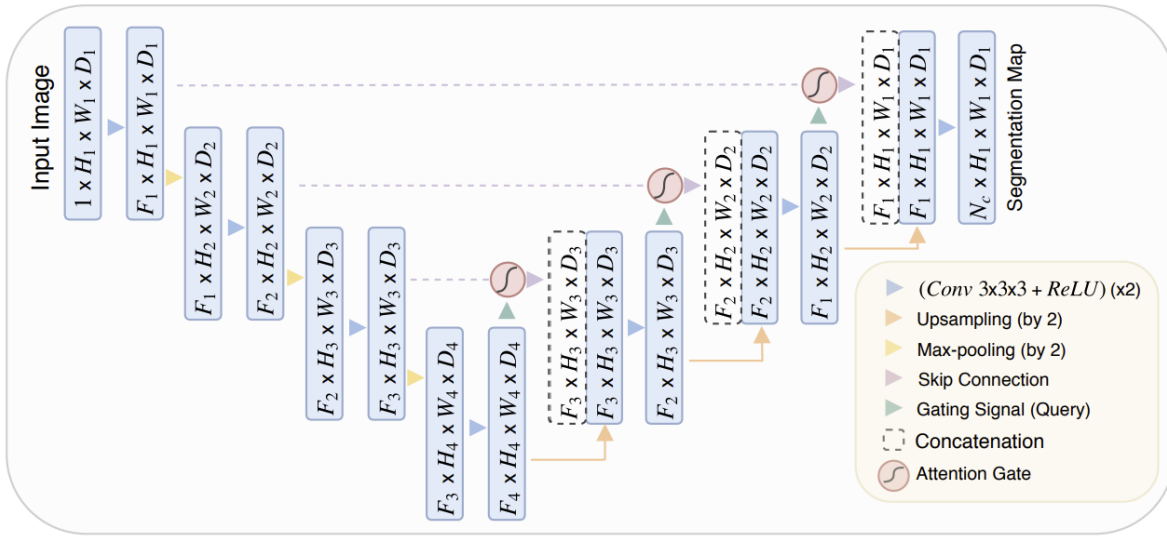


Figure 6.14 – Attention U-Net Architecture (Oktay et al., 2018)

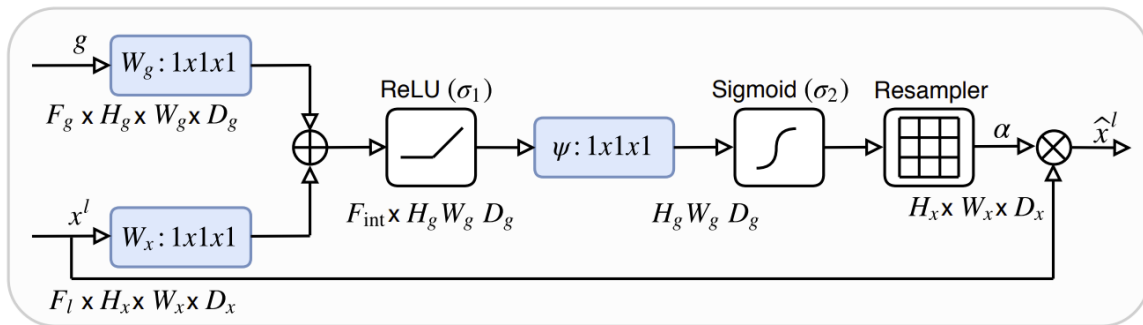


Figure 6.15 – Attention Gate used in Attention U-Net (Oktay et al., 2018)

6.5.1.2 Residual U-Net (ResUnet)

Residual U-Net (ResUnet) employs residual skip connections in the encoder and decoder layers of the network (Zhang et al., 2018). In the encoder part, the skip connection is added from the input to the output of the double convolution block. Similarly, in the decoder part, the output from the encoder layer is added to the output of the decoder layer (Fig. 6.16). The residual connection improves the gradient flow within the network during training phase, thereby avoids vanishing gradient problem. Zhang et al. (2018) compared performance of ResUnet and U-Net to segment roads in the field of remote sensing and obtained a precision of 91.8% using ResUnet while U-Net showed a precision of 90.5%.

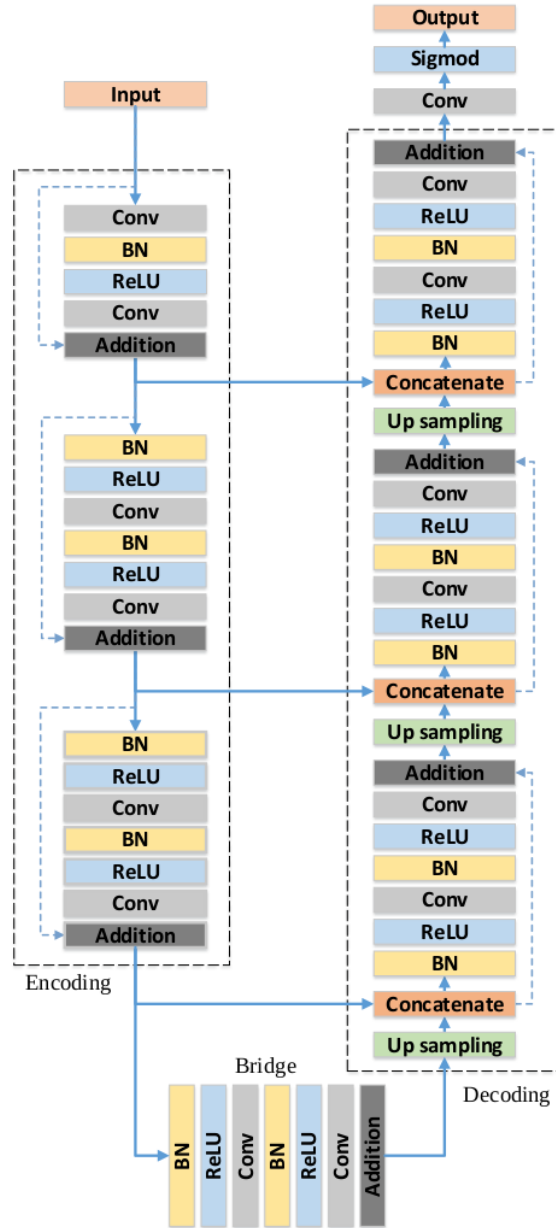


Figure 6.16 – Residual U-Net (ResUnet) Architecture (Zhang et al., 2018)

6.5.1.3 Transformer ResU-Net

Transformer ResU-Net (TransResU-Net) employs Resnet as backbone in the encoder layer of U-Net along with self-attention transformer (**Fig 6.17**). Transformers facilitate the capture of global contextual information (Tomar et al., 2022). Transformers consist of multi-headed self-attention modules (**Fig 6.18**), where they calculate the attention weights for the input. The transformer starts with the calculation of query, key and values, then the attention weights are calculated. Transformers were first used in language models (Vaswani et al., 2017), but recently they are used in image analyses. In language models, a transformer creates the token embedding for the words in a sentence using a token dictionary. As the position of a word in a sentence is important from grammatical and contextual perspective, transformers also generate positional embedding. Then these embeddings are used by multi-headed self-attention modules to calculate the dependency between different words. Then the output is forwarded to a feedforward network which predicts the next word in the sequence. For image data, a similar attention mechanism is applied to patches of the image instead of words in a sentence. The transformer model divides the image into smaller patches, flattens them, and applies the attention mechanism to obtain the relationships between these patches. In TransResU-Net, ResNet50 was proposed to be used as the backbone of the encoder part of the network.

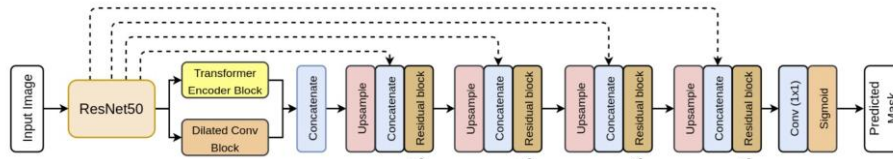


Figure 6.17 – Transformer ResU-Net (TransResU-Net) Architecture (Tomar et al., 2022)

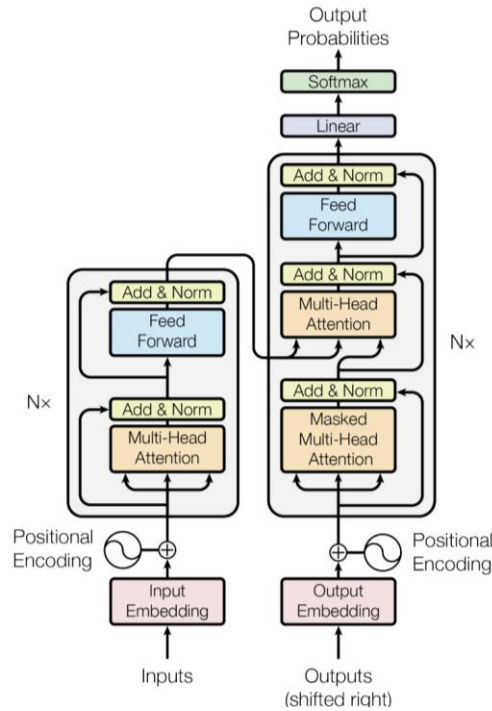


Figure 6.18 – Transformer (Vaswani et al., 2017)

6.5.1.4 Recurrent Residual U-Net (R2U-Net)

Recurrent Residual U-Net (R2U-Net) utilizes recurrent convolutional layers to improve the feature extraction and segmentation of the image (Alom et al., 2018). The recurrent convolutional layers were provided in the encoder layers, and the recurrent up-convolutional layers were provided in the decoder layers of the network (**Fig 6.19**). This recurrent residual convolution improves the quality of the features extracted from the image data. Alom et al. (2018) used R2U-Net to segment retina blood vessels in medical images.

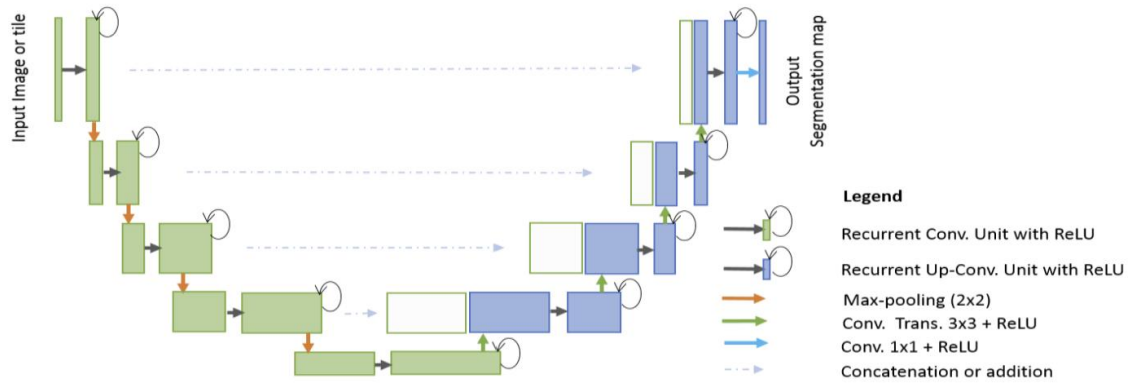


Figure 6.19 – Recurrent Residual U-Net (R2U-Net) Architecture (Alom et al., 2018)

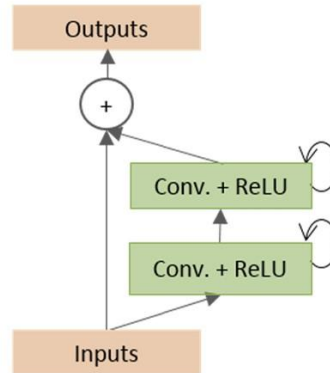


Figure 6.20 – Recurrent Residual Convolution Block used in R2U-Net Architecture (Alom et al., 2018)

6.5.1.5 ResUNet++

ResUNet++ (Jha et al., 2019) comprises a residual block, attention mechanism, and Atrous Spatial Pyramidal Pooling (ASPP). The attention blocks are integrated into the decoder block (**Fig 6.21**). The ASPP layer facilitates feature extraction at varying scales thus improving the accuracy of the network. In the ASPP layer (**Fig 6.22**), convolution kernel of 3×3 can be used but the dilation rate can be 1, 2, 3 or 6 (Chen et al., 2017). This dilation increases the receptive

field of the kernel. The different sizes of the receptive field help the network to obtain features at different scales.

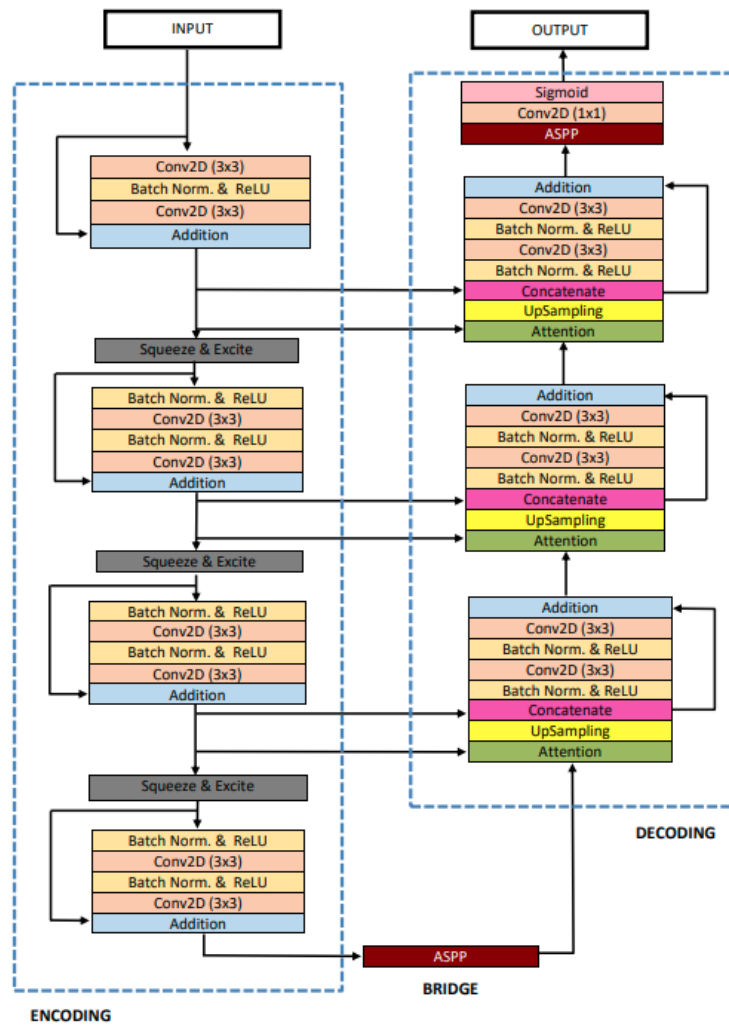


Figure 6.21 – ResU-Net++ Architecture (Jha et al., 2019)

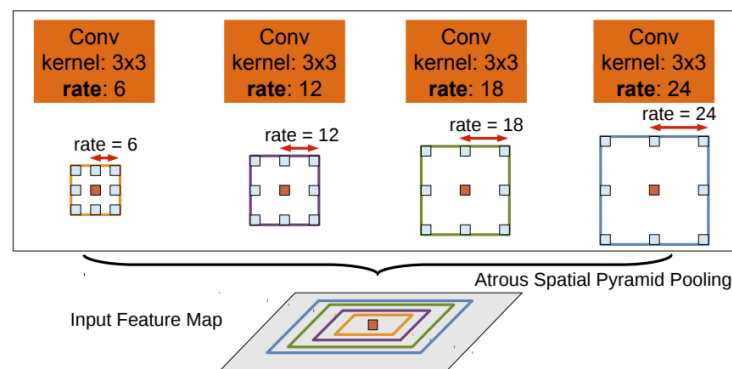


Figure 6.22 – Atrous Spatial Pyramidal Pooling (ASPP) (Chen et al., 2017)

6.5.2 Normalization Methods for Thermal Images

For the training of a neural network, the normalization of the image dataset helps with the optimization of network parameters during training and testing phases. In this study, seven different methods of normalization of the thermal image dataset were evaluated for different U-Net models. These normalization methods were based on the average temperature, standard deviation, minimum temperature, and maximum temperature.

Table 6.1 presents a summary of the thermal data normalization methods. In the first method, the images were normalized based on the average temperature and the standard deviation of individual image. In second and third methods, the minimum and maximum temperatures of the individual images were used to normalize each image. For the fourth method, the minimum and maximum temperature values were 0°C and 60°C, respectively. The thermal dataset had a maximum temperature of 60°C and a minimum temperature of 0°C. In the fifth method, mathematically there was no normalization as the average temperature was 0°C and standard deviation was kept at 1.0°C which does not change the temperature values of the image. In the sixth method, the average temperature of 38.5°C and a standard deviation of 3.5 were used which were based on the average temperature and standard deviation of thermal dataset. In the seventh method, the average temperature was kept the same as 38°C. However, the standard deviation was reduced to 0.5°C to observe the impact of a lower standard deviation on the results.

Table 6.1. Different Normalization Methods for Thermal Images

Parameter	Method-1	Method-2	Method-3	Method-4	Method-5	Method-6	Method-7
Mean Temperature	Variable per image	-	-	-	0°C	38.5°C	38.5°C
Standard Deviation	Variable per image	-	-	-	1°C	3.5°C	0.5°C
Minimum Temperature	-	Variable per image	Variable per image	0°C	-	-	-
Maximum Temperature	-	Variable per image	Variable per image	60°C	-	-	-

6.5.3 Neural Network Model Training on Thermal Dataset

PyTorch library was used to implement and train U-Net and its variants. The networks were trained using GPU (NVIDIA A100) of a high-performance computing cluster. The original thermal images had a resolution of 640×512 but the images were resized for the training of the neural networks. Different image sizes were tested (256×256, 352×352, and 512×512 pixels) and an image size of 352×352 pixels was selected as it provided reasonable accuracy and computational efficiency. Different batch sizes of 2, 4, 6, 8, and 10 were tested. It was found that the lower batch sizes caused noises in the predicted segmentation masks, therefore the batch size of 10 was selected. The number of epochs for training was set as 200. Out of 5,456 original and augmented images, 80% of the image dataset was used to train the models and 20% was used to test the performance of the neural networks. The initial learning rate was selected as 0.001 with an exponential decay over the time. Adam optimizer with a momentum value of 0.9 was used to perform the optimization during the back propagation within the training loops. The training and validation loss was monitored during the training loop.

Figures 6.23 to 6.28 show the training and validation loss for U-Net and its variants using the Method-1 of the thermal data normalization. Later in **Sec. 6.5.5**, the performance of each normalization method is discussed. The epoch with the lowest validation loss is also marked on the graph with dotted lines. The epoch that corresponds to the best performance for U-Net, Attention U-Net, ResUnet, TransResU-Net, R2U-Net, and ResUNet++ was 159, 67, 49, 64, 17, and 47, respectively. The validation loss at these epochs for U-Net, Attention U-Net, ResUnet, TransResU-Net, R2U-Net, and ResUNet++ had a value of 0.18, 0.28, 0.53, 0.24, 0.45, and 0.27, respectively. The models at best epochs were then used for further evaluation as discussed next.

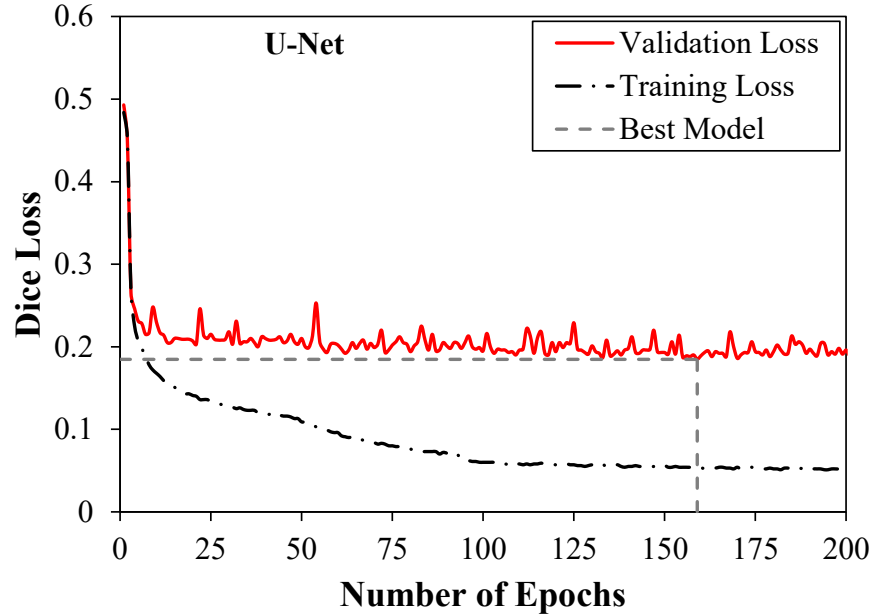


Figure 6.23 – Dice Loss of U-Net During Training and Validation on Thermal Dataset

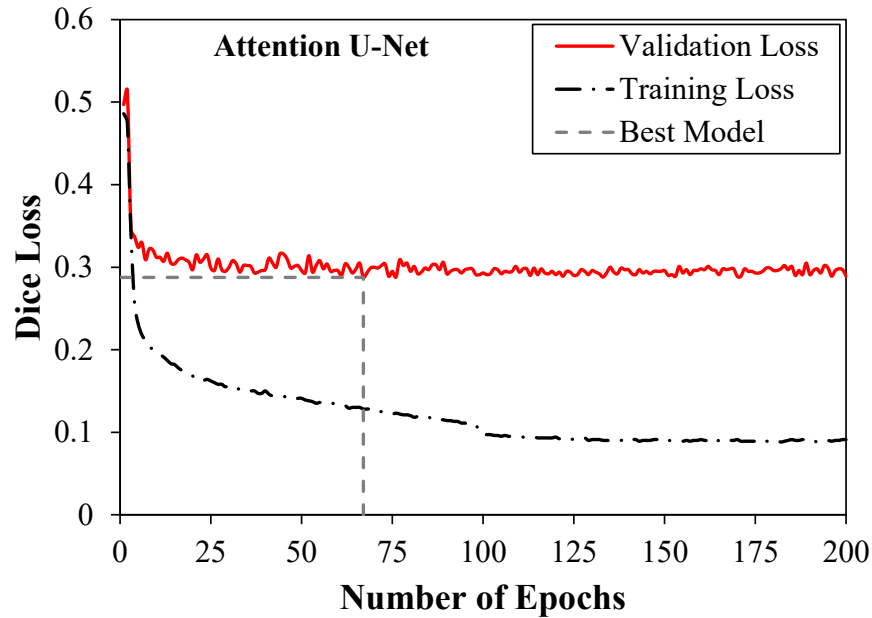


Figure 6.24 – Dice Loss of Attention U-Net During Training and Validation on Thermal Dataset

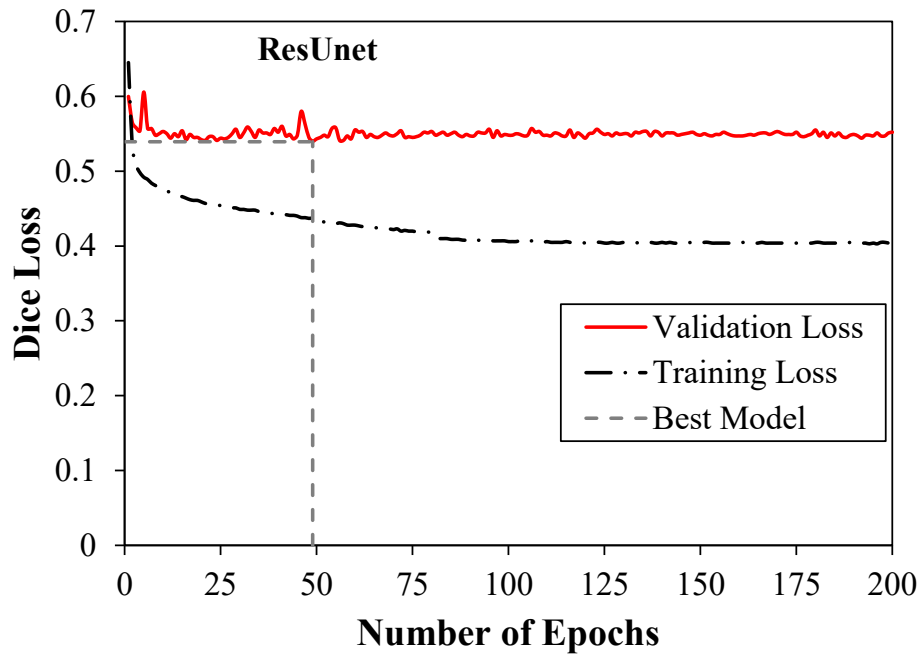


Figure 6.25 – Dice Loss of ResUnet During Training and Validation on Thermal Dataset

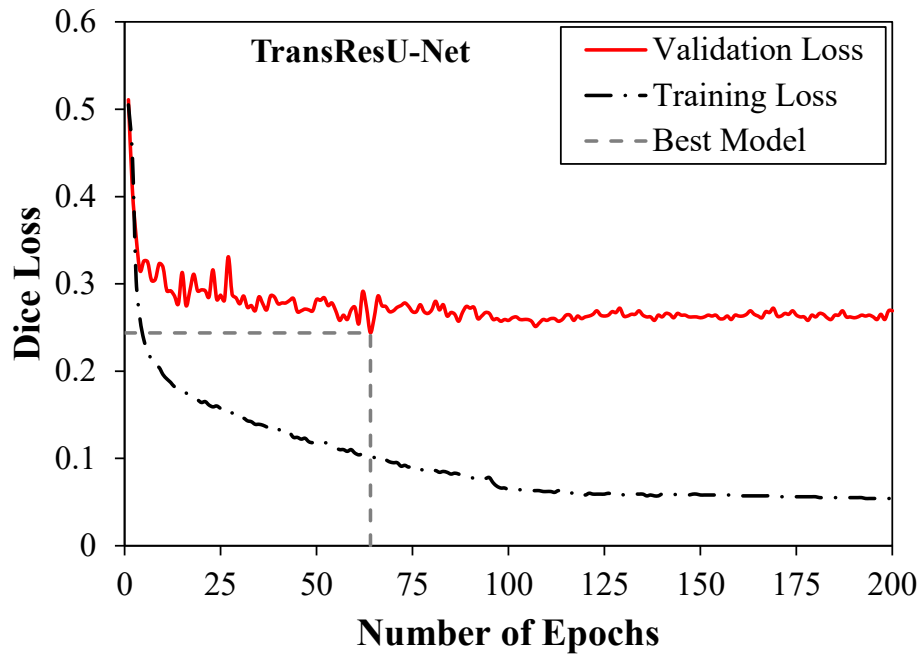


Figure 6.26 – Dice Loss of TransResU-Net During Training and Validation on Thermal Dataset

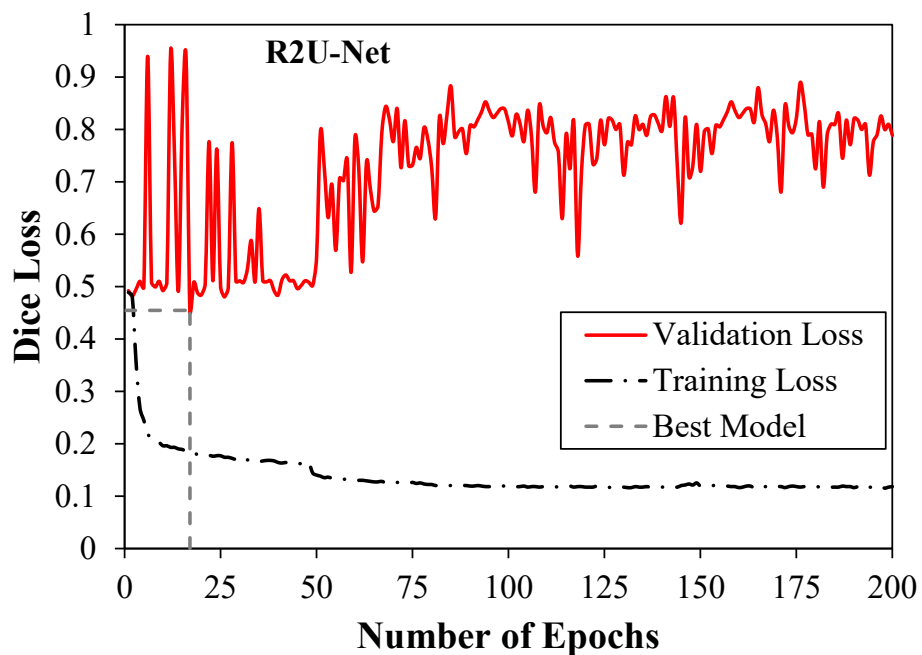


Figure 6.27 – Dice Loss of R2U-Net During Training and Validation on Thermal Dataset

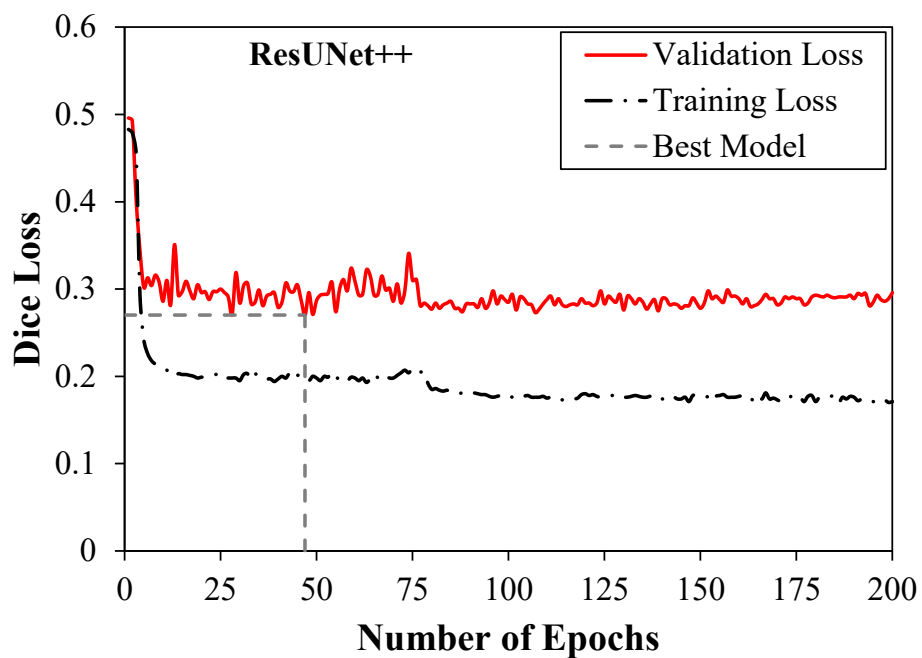


Figure 6.28 – Dice Loss of ResUNet++ During Training and Validation on Thermal Dataset

6.5.4 Performance Indices of Neural Networks on Thermal Images

The validation dataset was used to assess the performance of the neural networks in detecting concrete delamination from thermal images. To evaluate the performance of different neural networks, five different performance indices including the accuracy, precision, recall, F1 score, and Jaccard index were calculated as:

$$\text{Accuracy:} \quad \frac{TP + TN}{TP + TN + FP + FN} \quad (6.2)$$

$$\text{Precision:} \quad \frac{TP}{TP + FP} \quad (6.3)$$

$$\text{Recall:} \quad \frac{TP}{TP + FN} \quad (6.4)$$

$$\text{F1 Score:} \quad \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6.5)$$

$$\text{Jaccard Index:} \quad \frac{A \cap B}{A \cup B} \quad (6.6)$$

where, TP is the true positive, FP is the false positive, TN is the true negative, and FN is the false negative. If the pixel of an image belongs to a delaminated region in the ground truth and the model also accurately predicts it as a part of the delaminated region, then this will be labeled as a TP value. For a solid region of concrete (no delamination), if both ground truth and prediction indicate that the pixel belongs to the solid region, then it is called TN. The result from models will be FP if the model inaccurately predicts a solid region as delaminated. If the model misses any delaminated pixel and inaccurately predicts it as a solid region then it is called FN. These four parameters (TP, TN, FP, and FN) are then used to calculate the indices of **Eq. 6.2 to 6.6**. The five performance indices can range from 0 to 100%, where a higher value indicates a better performance for the network.

The accuracy (**Eq. 6.2**) indicates the accuracy level of the model in identifying the delaminated and solid (no delamination) regions (e.g., how many delaminated and solid pixels are accurately identified). For an unbalanced dataset such as the one that exists in the present study, the accuracy might be high and might not be a true representative of the performance of the network. The precision considers the positive values predicted by the network and measures the ratio of the accurate positive values to the sum of all positive values. In other words, this performance index reveals that out of all the predicted delaminated pixels, how many truly belong to the delaminated regions. On the other hand, the recall penalizes the model by including FN if it misses the delaminated regions. The F1-score depends on the precision and recall of the network. The high values for precision and recall will translate into a high value for the F1-score. The Jaccard index provides information regarding the overlap between the actual and predicted delaminated regions in the image.

6.5.5 Effects of Different Thermal Data Normalization Methods on Network Performance

Table 6.2 presents the performance of seven different normalization methods to predict delamination in thermal images using U-Net. These performance parameters are based on 419 validation images. **Figure 6.29** shows the performance of each normalization method. Method-1, where images were normalized based on the average temperature and the standard deviation of individual images, performed better in terms of accuracy (86%) in predicting delamination in thermal images. In general, the normalization methods based on individual image values (Methods 1, 2 and 3) performed better than the normalization values based on the complete dataset. Based on these findings, Method-1 was used to normalize the image dataset and to compare the performance of U-Net and its variants.

Table 6.2. Performance of U-Net with Different Normalization Methods for Thermal Data

Parameter	Method-1	Method-2	Method-3	Method-4	Method-5	Method-6	Method-7
Accuracy	86.2	86.1	85.9	84.3	81.9	24.2	40.8
Precision	69.7	69.6	69.2	65.9	60.6	23.8	24.3
Recall	75.8	73.1	74	70.6	63.4	26.3	28.1
F1 score	67.4	66.9	67.2	63.5	57	36.1	32.6
Jaccard index	54.5	53.9	54.1	50.1	43.4	23.8	21.6

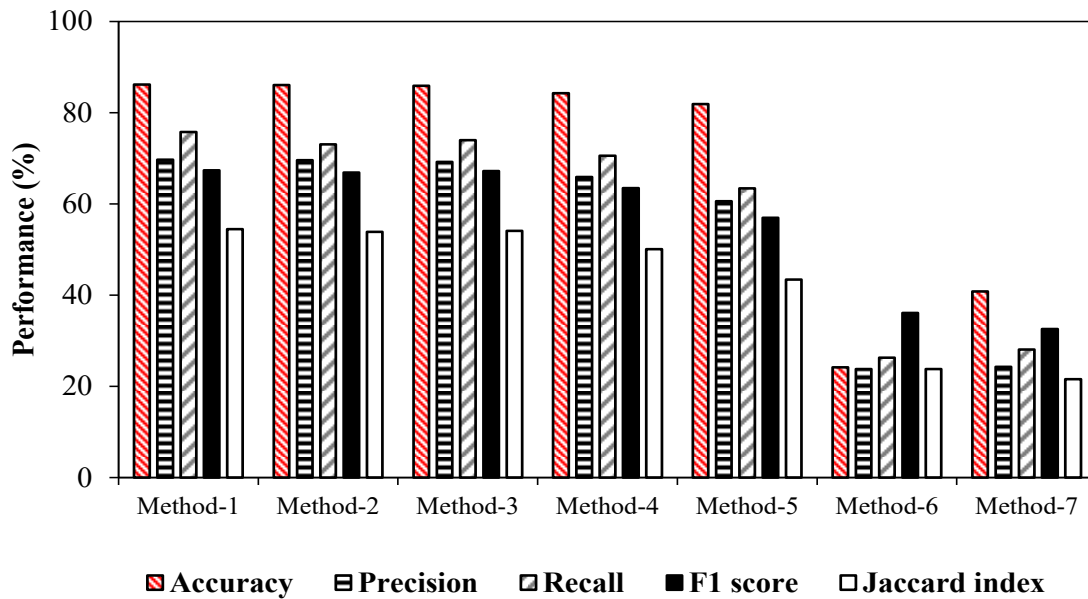


Figure 6.29 – Performance of U-Net with Different Normalization Methods for Thermal Images

Figure 6.30 shows sample results for each normalization method along with the thermal image and its ground truth. The normalization based on the mean temperature and standard deviation of individual image (Method-1) was able to capture the precise location and general shape of delaminated regions in the thermal images. The results of Methods 6 and 7, where the single value of minimum and maximum temperatures were used to normalize the thermal images, had the lowest performance. Method-6 predicted that all the pixels in the image were delaminated, which was not correct. Based on the performance parameters and sample predicted results for each normalization method, Method-1 was found as the best approach and was used in the software development phase.

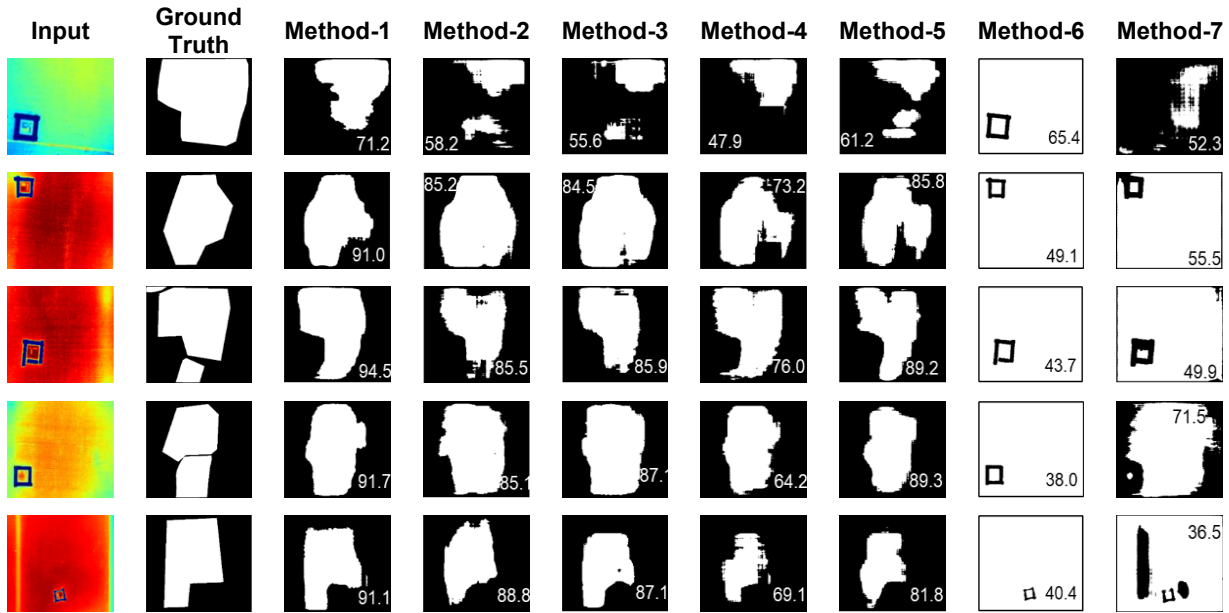


Figure 6.30 – Sample Delamination Detected by U-Net Using Different Normalization Methods for Thermal Images

6.5.6 Performance of Neural Networks on Delamination Detection in Thermal Images

Table 6.3 presents a summary of the performance of U-Net and its five variants in detecting delamination in thermal images. These performance parameters are based on 419 validation images. U-Net outperformed the other variants in terms of all performance parameters (**Fig. 6.31**). For the U-Net model using all the images of the validation set, the accuracy was 86.2%, the precision was 69.7%, the recall was 76%, the F1-score was 67.4%, and the Jaccard index was 54.5%. Attention U-Net was ranked second with an accuracy of 85.9%, a precision of 60.1%, a recall of 56.3%, an F1-score of 58.1%, and a Jaccard index of 54.1%. ResUNet++ was ranked third with an accuracy of 85.8%, a precision of 60.8%, a recall of 55.1%, an F1-score of 57.6%, and a Jaccard index of 53.5%. TransResU-Net had an accuracy of 83.6%, a precision of 44.4%, a recall of 30.9%, an F1-score of 34.1%, and a Jaccard index of 24.3%. ResUnet had an accuracy of 78.8%, a precision of 10.6%, a recall of 47.6%, an F1-score of 17.3%, and a Jaccard index of 10.3%. R2U-Net performance was the worst among all models with an accuracy of 11.2%, a precision of 2.6%, a recall of 1.1%, an F1-score of 1.5%, and a Jaccard index of 0.8%.

Table 6.3. Performance Parameters for Different Variants of U-Net Architecture

Parameter	U-Net (%)	Attention U-Net (%)	ResUnet (%)	TransResU-Net (%)	R2U-Net (%)	ResUNet++ (%)
Accuracy	86.2	85.9	78.8	83.6	11.2	85.8
Precision	69.7	60.1	10.6	44.4	2.6	60.8
Recall	75.8	56.3	47.6	30.9	1.1	55.1
F1 score	67.4	58.1	17.3	34.1	1.5	57.6
Jaccard index	54.5	54.1	10.3	24.3	0.8	53.5

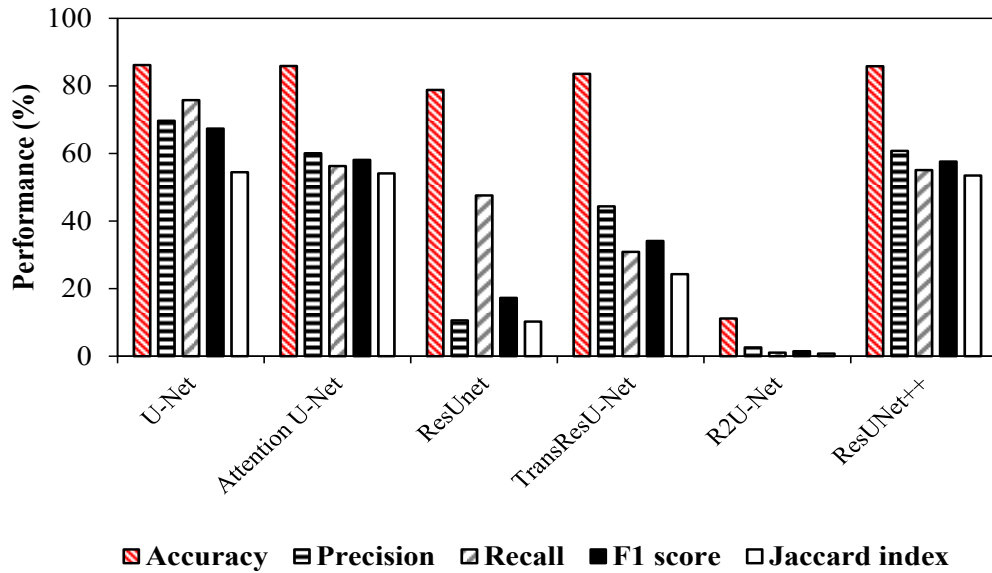
**Figure 6.31 – Performance of Different Variants of U-Net Architecture in Detecting Delamination in Thermal Images**

Figure 6.32 shows sample results for all variants of U-Net as well as the original thermal images and their ground truth. It can be seen that U-Net performs better compared with the other neural networks. U-Net was able to capture the shape and precise location of delaminated regions in the thermal images. Attention U-Net was able to obtain the general shape and precise location of delaminated regions; however, some pixels related to delamination were missed. TransResU-Net underestimated the delaminated regions, even missed some portions of delamination, and was not able to generate correct shape of the delaminated regions. R2U-Net missed large delaminated regions and was found to be the least accurate model. In R2U-Net, recurrent convolution is used that might have suppressed the delaminated regions instead of making them more prominent. As discussed earlier, the original R2U-Net was developed by Alom et al. (2018) for segmentation of retina blood vessels in medical images. The blood vessels were prominent and exhibited high contrast against the background, thus the recurrent convolutional block facilitated feature extraction. However, in the case of delamination in thermal images, the boundary between delaminated and adjacent regions is not distinct, leading the recurrent convolutional blocks to suppress features, resulting in poor performance. In the case of

ResUNet++, the boundaries of predicted delaminated regions were not refined, and model underestimated the delaminated regions. In general, the U-Net model performed better in obtaining precise location and general shape of the delaminated regions in thermal images. Based on these findings, the U-Net model was selected for further evaluation next, in which samples for the highest and lowest accuracies and performance on a bridge deck are presented.

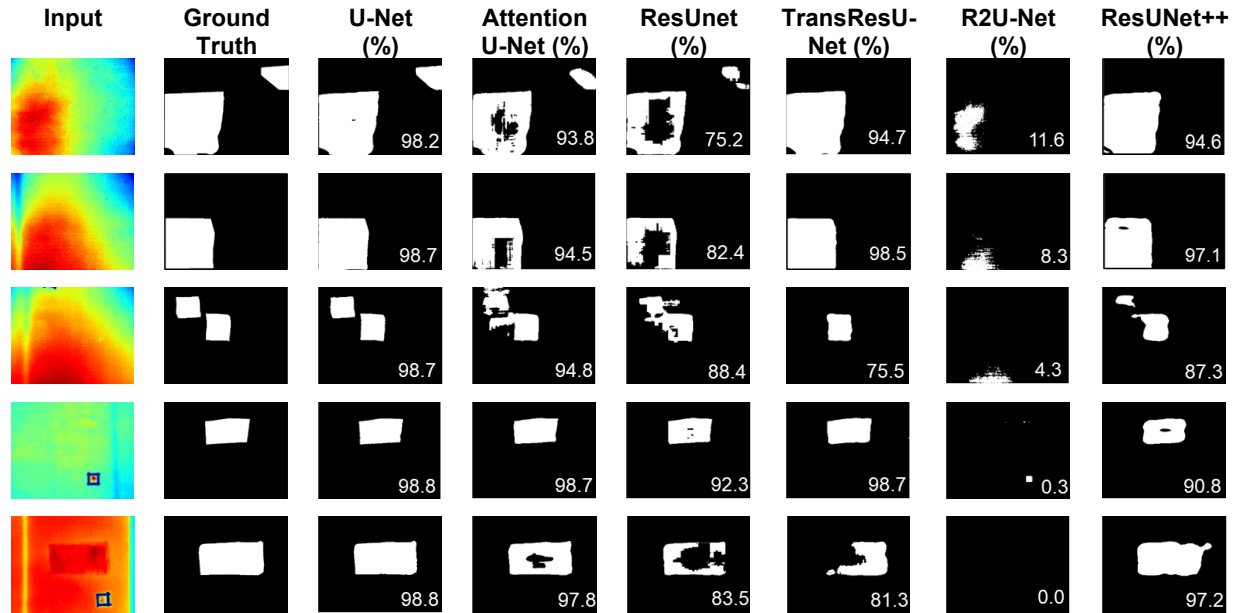


Figure 6.32 – Sample Delamination Detection by Different U-Net Models in Thermal Images

6.5.7 Samples of Delamination Detections in Thermal Images Using Selected Model

Figure 6.33 shows samples of delamination detections using the top model (U-Net) and normalization method when the performance of the model was high for validation image dataset. Furthermore, Fig. 6.34 shows sample detections when the performance indices were low. In both figures, the left image is the original thermal image that was fed to the neural network, the middle image is the ground truth, and the right image shows the predicted delaminated regions from the model. The accuracy of the U-Net model in some cases exceeded 90% (e.g., Fig. 6.33); however, poor predictions with a low accuracy of 56% (e.g., Fig. 6.34) were also observed. Note that the detections shown in Fig. 6.33 and 6.34 are the best and worst findings of the study. In most cases, U-Net overestimated the delaminated areas when compared with the ground truth thus reducing the performance measures of the model. The samples with the lowest accuracies as shown in Fig. 6.34 predicted a higher area for delamination while the actual delaminated area was smaller indicating that the top model is conservative in detecting the delaminated areas of bridge decks.

Overall, U-Net was successful in predicting the location and area of the delaminated regions over the evaluation set.

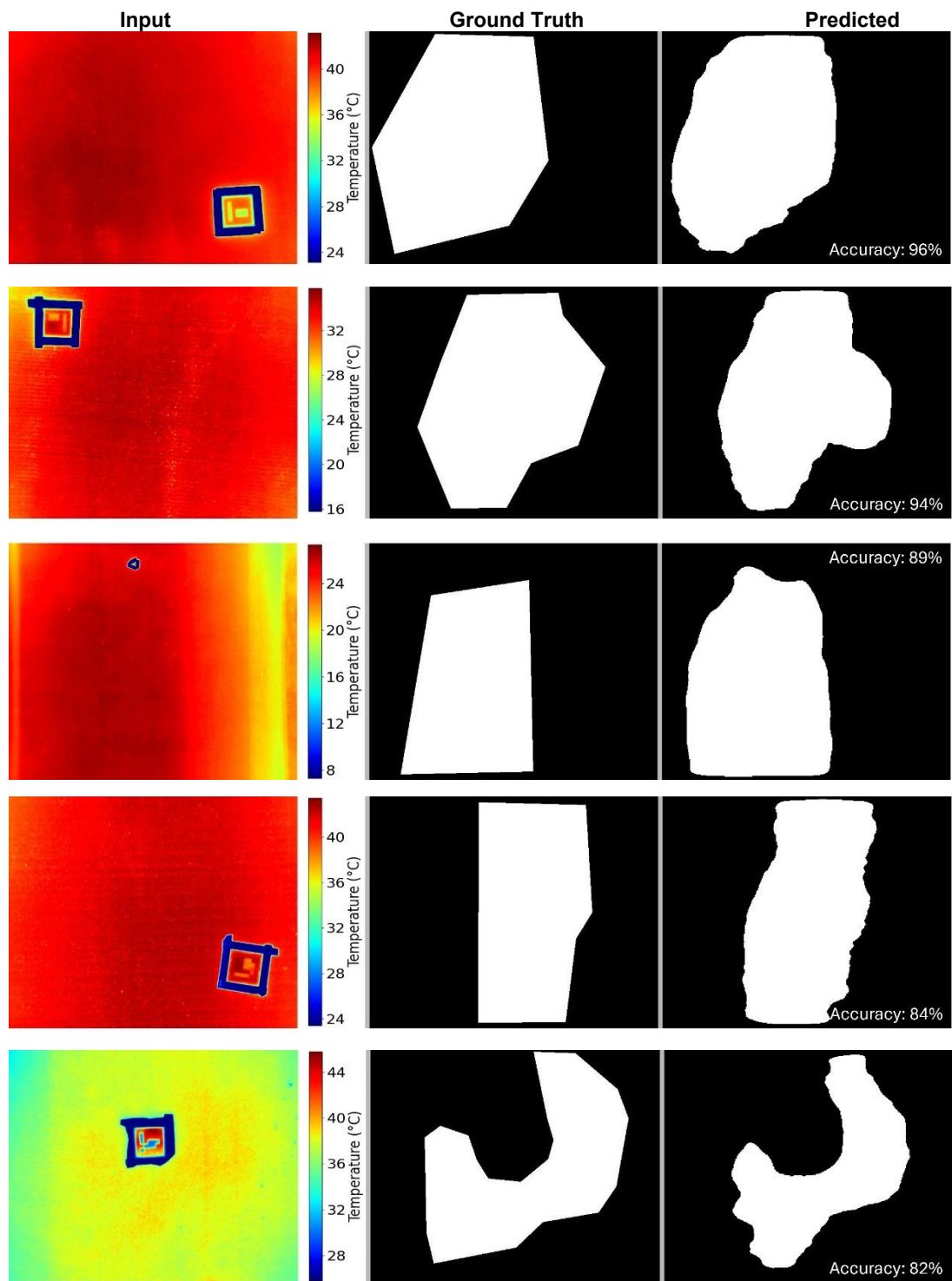


Figure 6.33 – High Performance of Select Neural Network in Detecting Delamination in Thermal Images

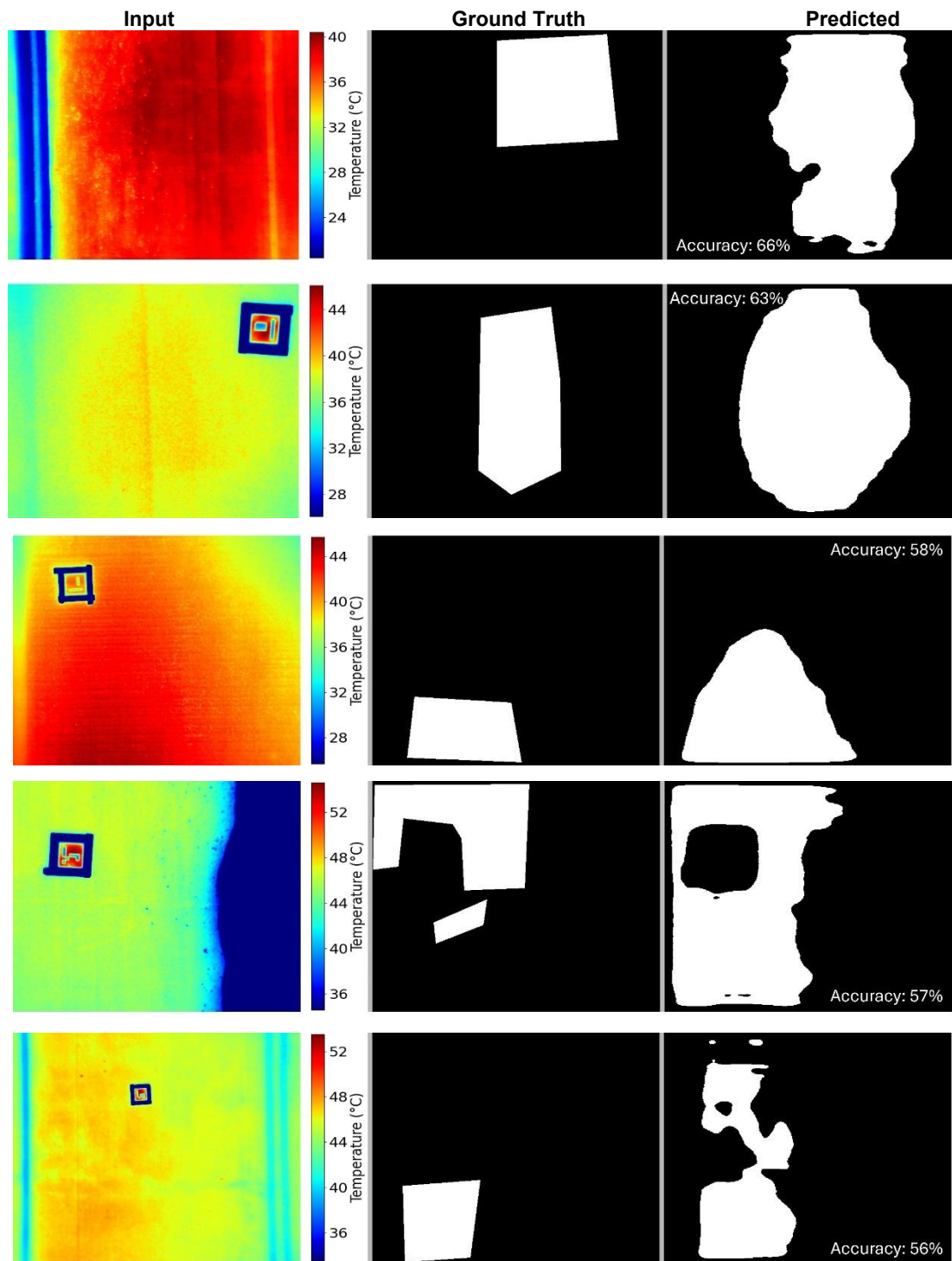


Figure 6.34 – Low Performance of Select Neural Network in Detecting Delamination in Thermal Images

Figure 6.35 shows individual RGB images of a bridge and their corresponding, overlaid thermal images. The outcome of the U-Net model (white areas of the image) is also included and superimposed on each image. The model accurately identified the hot areas of the deck as

delamination, similar to a human inspector. Furthermore, it successfully ignored false positives, such as hot expansion joints and railings, by not marking them as delamination.

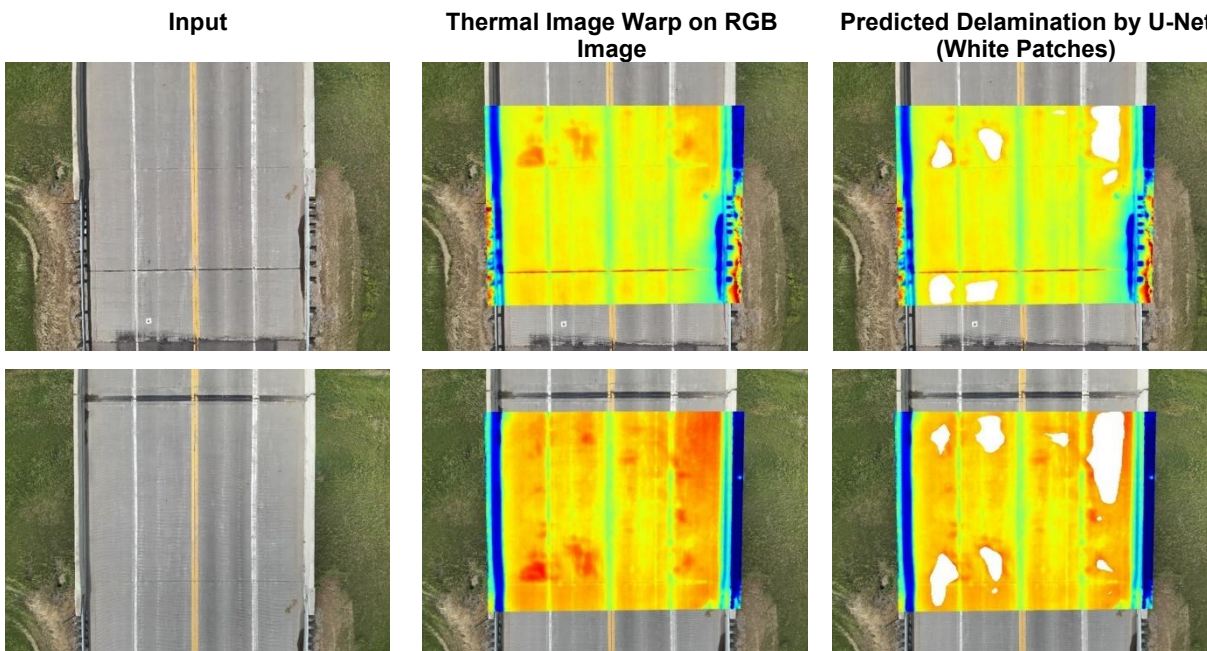


Figure 6.35 – Performance of Select Neural Network in Detecting Delamination in Individual Thermal Images Superimposed on RGB Images

6.6 Samples of Delamination Detection on Thermal Maps of Bridge Decks

A novel approach to detect delamination of concrete bridge decks using AI and thermal imaging was presented in Sec. 6.2. Following the same methodology, this section provides sample results in which the selected neural network, U-Net, was used to analyze thermal maps of bridge decks to develop delamination maps.

After stitching the thermal images for the full length of each bridge (e.g., **Fig. 6.36**), the next step is to analyze the thermal information and detect delaminated regions. The best U-Net model, as discussed in the previous sections, can be used in this task. First, small patches are extracted from the bridge 2D thermal map and are individually fed to the neural network. A patch with a width (in the bridge transverse direction) that was equal to the width of bridge lanes (e.g., usually 10-13 ft (3-4 m)) and a height (in the bridge longitudinal direction) of 500 pixels was found suitable. Each successive patch (moving them in the bridge longitudinal direction) can have an overlap of 50% with the previous patch. For example, a total of 36 patches were obtained for the bridge shown in **Fig. 6.36**. This bridge was 293 ft (89 m) long and had two lanes each 12 ft (3.65 m) wide. These individual patches are fed to the U-Net model and delamination masks for each patch are obtained. The 2D delamination map of the whole bridge can be reconstructed from the predicted mask generated for each patch.

Figure 6.36 shows a sample result in which the white color represents the detected delaminated regions of the bridge deck. The figure also includes a corresponding ground truth map for this

bridge, which was obtained from the chain-drag method by an expert inspector. Compared with the ground truth, the model showed an accuracy of 77% on the intersection over union. As a global performance assessment, it was found that the total physical delaminated area (e.g., in²) of the prediction was 99% of that of the ground truth. Note that the thermal image showed higher temperatures at both ends of the bridge. This was because vaulted (mostly hollow) abutments were used at the ends. Overall, the neural network was reasonably accurate and was able to detect delamination, their location and areas on the entire bridge deck.

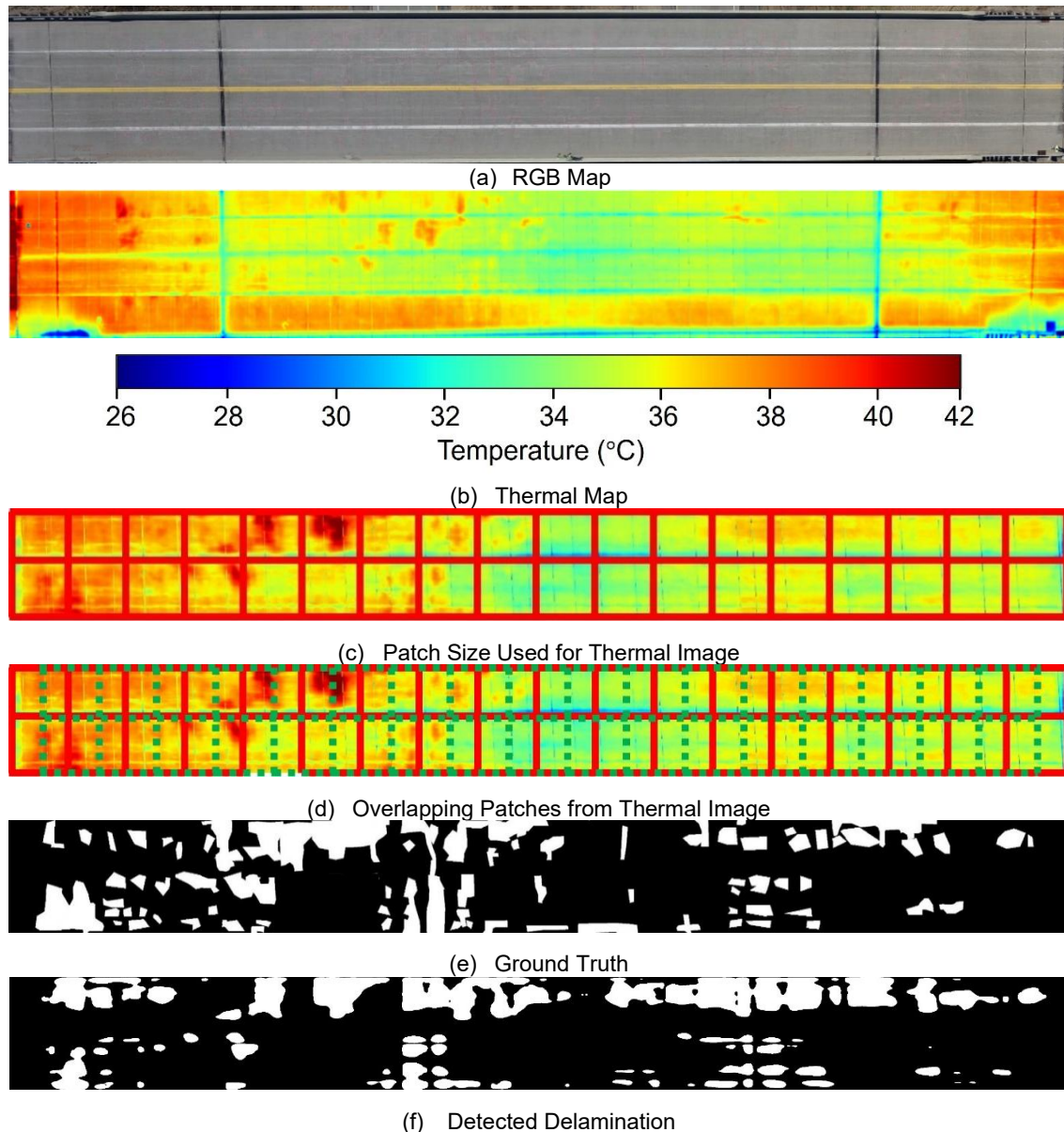


Figure 6.36 – Performance of U-Net in Detecting Delamination of an 89.3 m (293 ft) Long Bridge Deck

The same methodology to collect thermal data, to generate 2D RGB-thermal maps, and to analyze them using U-Net was successfully exercised on several other bridges. **Figures 6.37 to 6.42** show the results for six bridges. Unfortunately, the ground truth (using a chain-drag method) was not available for the entire decks of these six bridges. Nevertheless, the 2D thermal maps can be used as the reference to check if U-Net is able to find defected areas. A ground truth solely based on the observed thermal images was generated by the research team to better understand the performance of U-Net on these bridges. It is understood that this method is subjective and not in line with practice but was done to comment on the model's performance. In **Fig. 6.37**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 83% based on the intersection over union, and the global accuracy (area to area) was 99%.

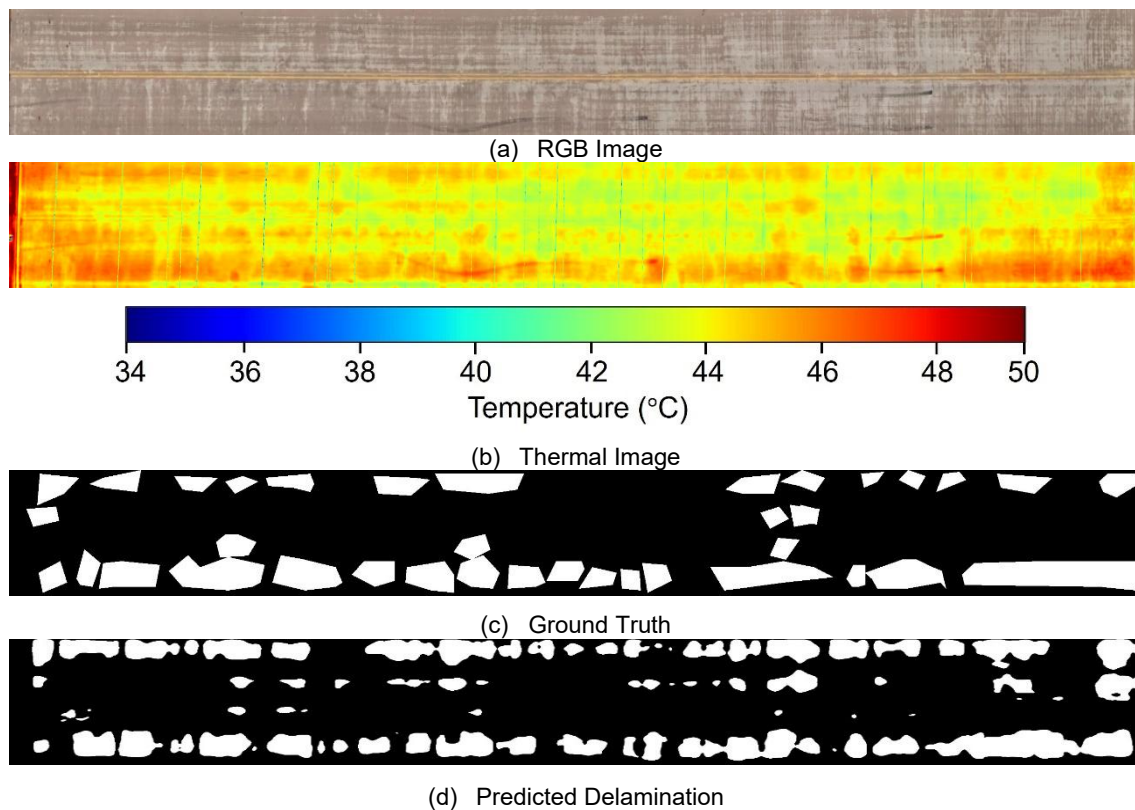


Figure 6.37 – Performance of U-Net in Detecting Delamination of a 77.4 m (253.9 ft) Long Bridge Deck

In **Fig. 6.38**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 81.5% based on the intersection over union, and the global accuracy (area to area) was 97%.

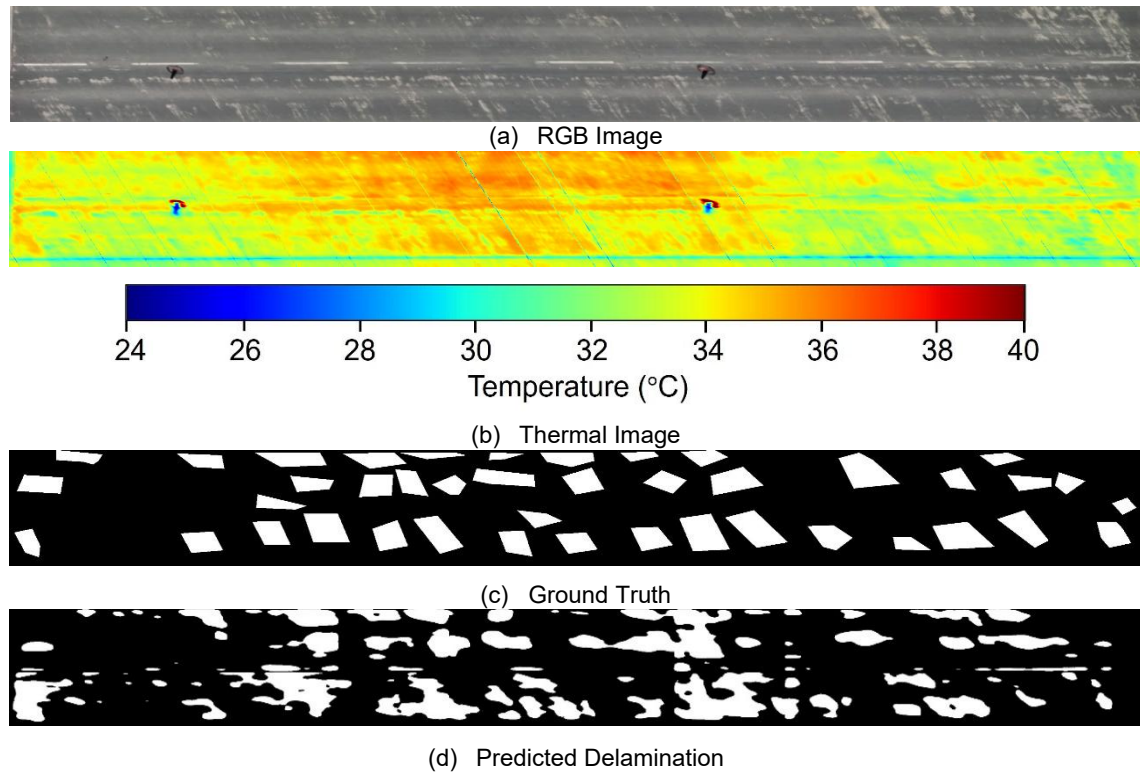


Figure 6.38 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Long Bridge Deck

In **Fig. 6.39**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 81% based on the intersection over union, and the global accuracy (area to area) was 88%.

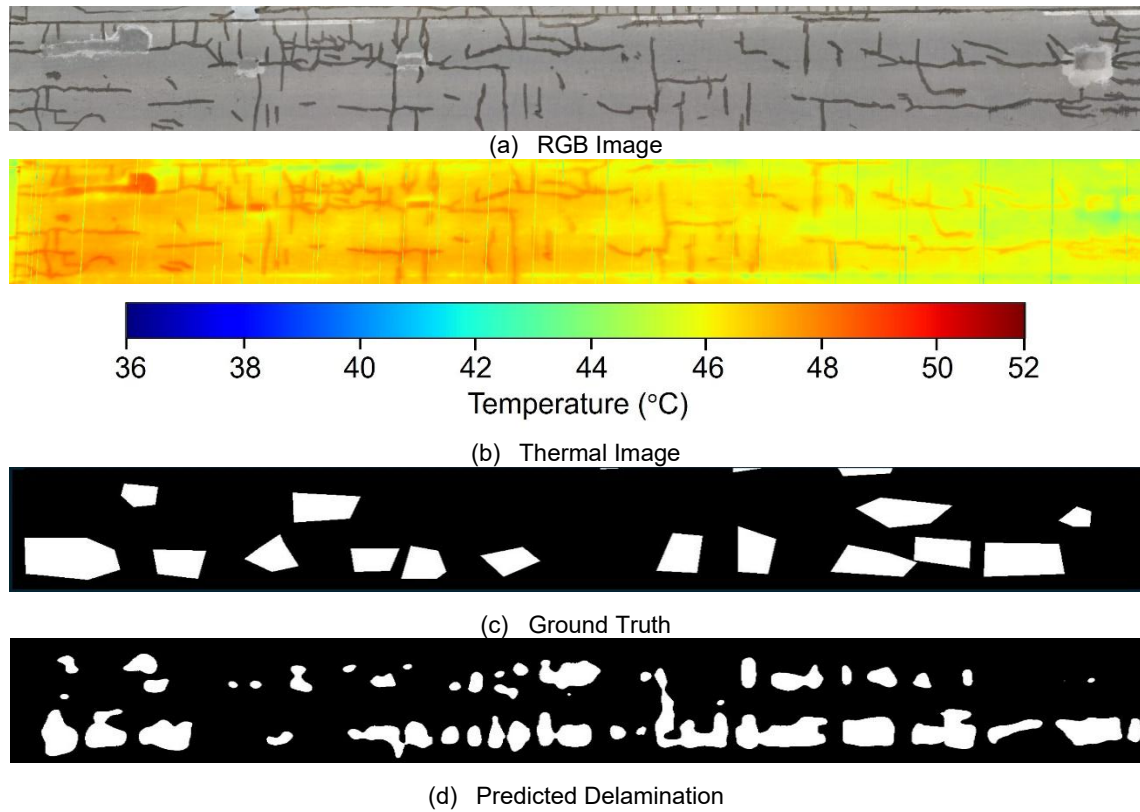


Figure 6.39 – Performance of U-Net in Prediction Delamination on 36.3 m (119.1 ft) Bridge Deck

In **Fig. 6.40**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 83% based on the intersection over union, and the global accuracy (area to area) was 97%.

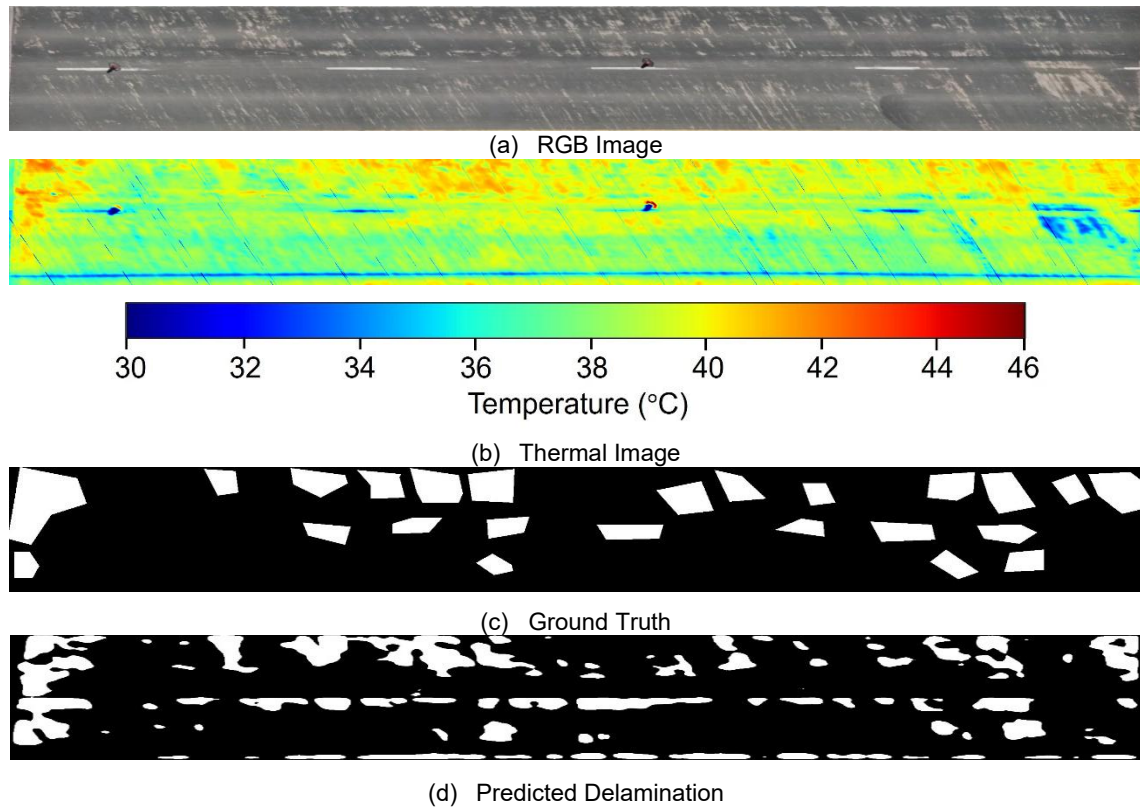


Figure 6.40 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Long Bridge Deck

In **Fig. 6.41**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 83% based on the intersection over union, and the global accuracy (area to area) was 95%.

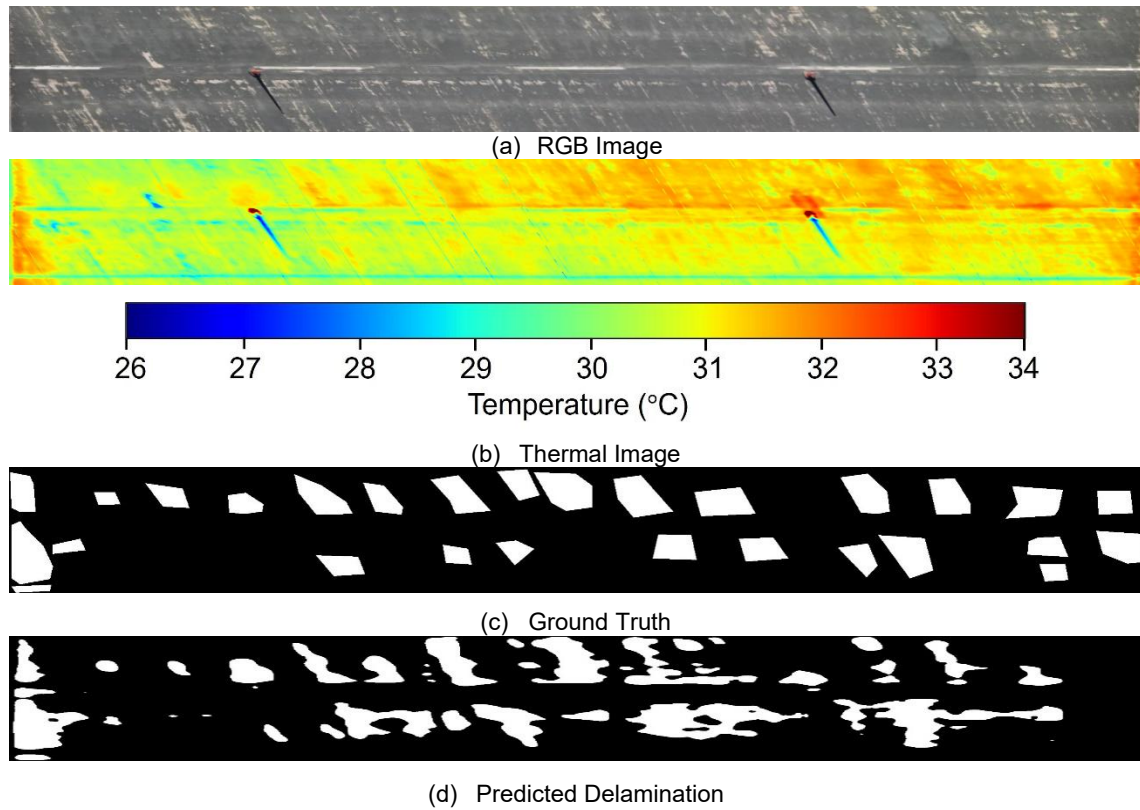


Figure 6.41 – Performance of U-Net in Detecting Delamination of a 52.2 m (171.3 ft) Bridge Deck

In **Fig. 6.42**, U-Net predicted the delamination in the concrete bridge deck with an accuracy of 82% based on the intersection over union, and the global accuracy (area to area) was 97%.

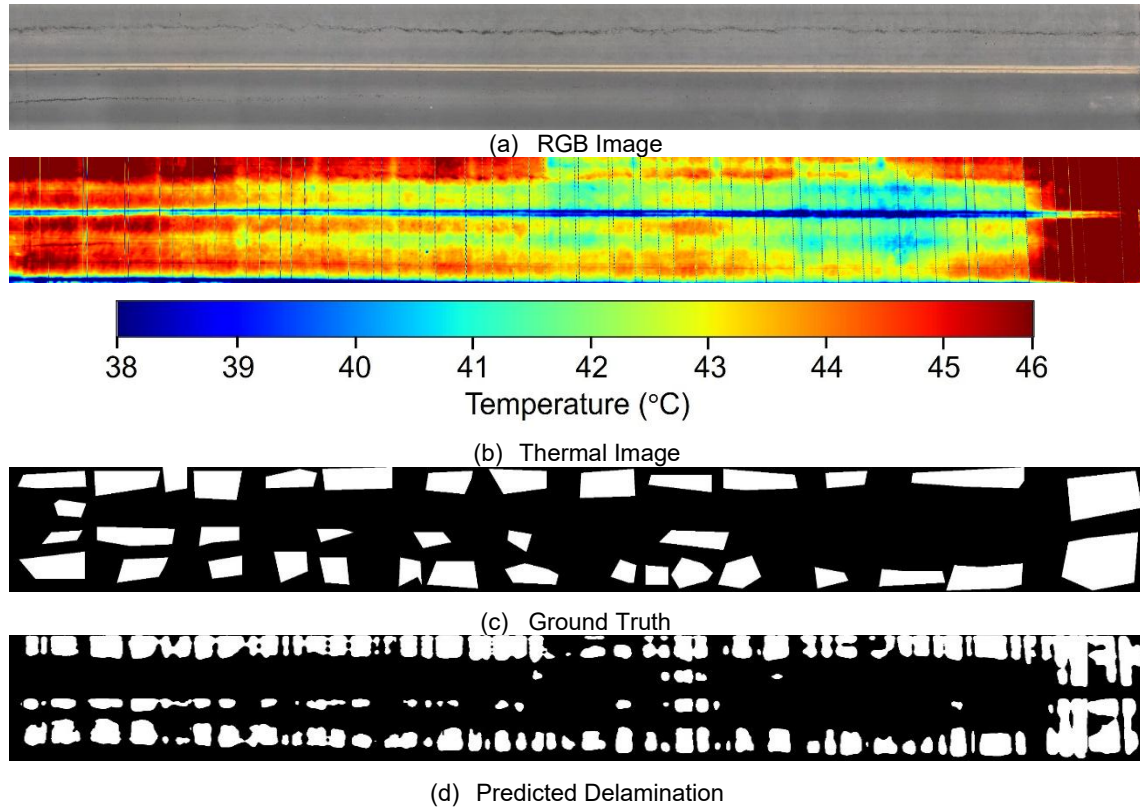


Figure 6.42 – Performance of U-Net in Detecting Delamination of a 77.4 m (253.9 ft) Long Bridge Deck

In summary, the proposed thermal map generation method, the selected neural network for thermal data analysis, and the thermal data normalization method were found to be viable methods of thermal image processing for detecting delamination of concrete bridge decks. An area-to-area accuracy of 88% to 99% with an average accuracy of 96% was achieved for delamination detection across the seven bridges presented in this study.

CHAPTER 7 – RGB BASED STEEL GIRDER CONDITION ASSESSMENT WITH A HUMAN-IN-THE-LOOP METHOD

7.1 Introduction

Paint loss is a common type of damage observed in steel girders, often resulting from environmental exposure and aging. This damage is visible and can be identified using RGB imaging. In this study, a simple approach is proposed to assist bridge inspectors in assessing such damage. Inspectors can capture RGB images of steel girders, which are then corrected for lens distortions and perspective errors. The corrected images are then manually annotated by the user for visible defects. Using a known physical scale, the annotated damage can be quantified in terms of physical area. As a result, the process enables bridge inspectors and users to accurately estimate the defected areas of steel girders.

7.2 Image Warping and Scaling

High-resolution images of steel girders can be acquired using either drone-mounted RGB cameras or conventional handheld cameras. Among these, drone-based imaging offers a distinct advantage in terms of accessibility and viewpoint control, enabling the capture of orthonormal images that reduce perspective distortion. This is particularly useful when inspecting girders located in hard-to-reach areas or at elevated heights. However, even when images are taken at oblique angles whether due to site constraints or camera positioning, they can still be geometrically corrected through image rectification. This involves warping the image onto a reference plane so that the imaged surface aligns with a desired view.

Figure 7.1 shows the proposed workflow for the inspection of steel girders. An important preprocessing step in this workflow is the lens distortion correction, which compensates for nonlinear distortions introduced by the camera lens, such as barrel or pincushion distortion. These distortions can significantly affect spatial accuracy if not addressed. The process for computing and applying distortion coefficients is discussed in detail in **Ch. 5**. Once distortions are removed, the resulting undistorted image can be used to obtain damage map after geometrical transformations.

To warp the corrected image onto a reference plane, the user manually selects four key points in the image that correspond to known locations on the girder typically at the corners or along the edges. These points serve as inputs to a homography transformation, allowing the image to be rectified so that girder appears as if viewed directly from the side, perpendicular to its longitudinal axis. The physical dimensions of the girder, such as its length and width, are typically available from structural drawings or can be measured during field inspections. Another option is the height and distance of steel girder stiffeners. By aligning these known physical measurements with their corresponding pixel distances in the image, a pixel to physical scale ratio can be estimated.

This scale calibration enables accurate spatial quantification within the image. Using the rectified and scaled image, inspectors can annotate regions of interest such as areas with paint loss, corrosion, or other surface defects. The annotated regions can then be quantified in terms of physical area, providing an objective and measurable estimate of the extent of damage. This approach enhances the precision of visual inspection of the steel girders for condition

assessment. Note that since the user is involved in the process, such technique is called a “human-in-the-loop” method in the field of Artificial Intelligence (AI).

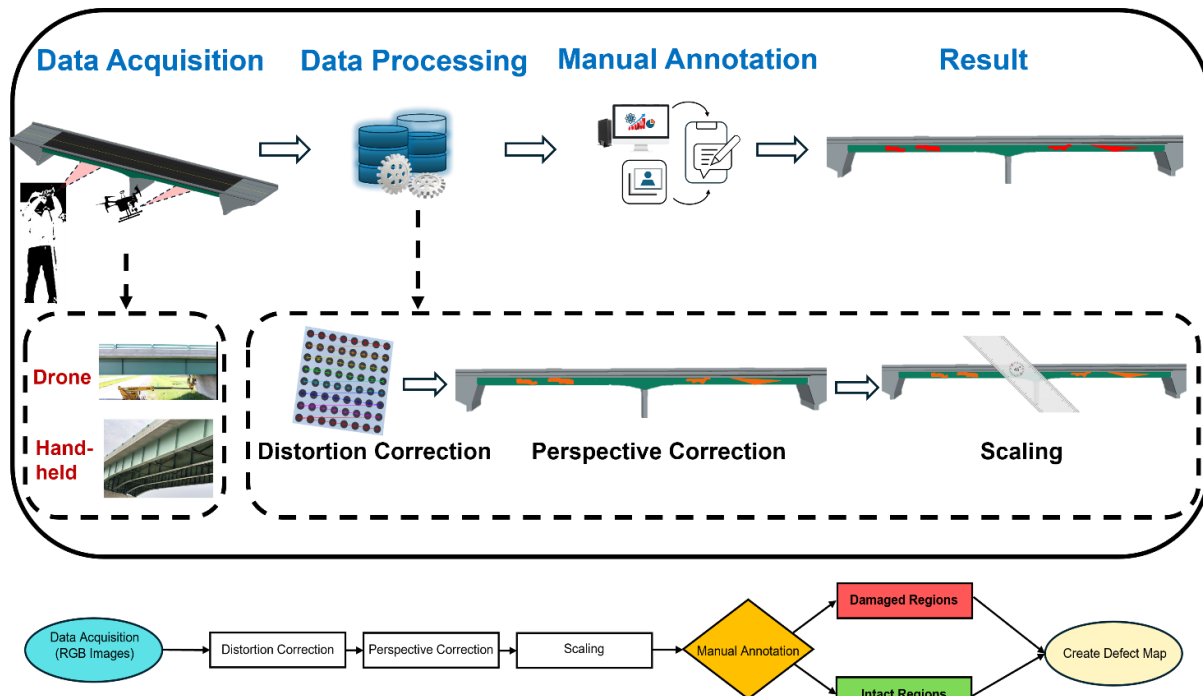


Figure 7.1 – Workflow for Steel Girder Defect Quantification Using a Human-in-the-Loop Method

7.3 Manual Defect Annotation for Steel Girders

To illustrate the process, consider a steel girder of a bridge shown in **Fig. 7.2a**. A conventional camera was used to capture the image. Since the image was not captured at a 90° angle to the girder’s web, thereby the perspective and lens distortions should be minimized. The lens distortion was removed using the camera calibration parameters. The corrected image was then used to correct the perspective of the girder by selecting four points along the girder (**Fig. 7.2b**). From the field, the distance between the four points and the depth of the girder (0.92 m or 3 ft) were measured. Using this information, the image view was corrected, and pixel-to-length ratio was obtained. Subsequently, the user manually annotated the defected regions (paint loss herein) of this steel girder, and the software generated a damage map with a condition state estimation per defect type (as defined in **Ch. 3**). For this example, the annotated region with paint loss had an area of 2.5 ft² (0.23 m²).

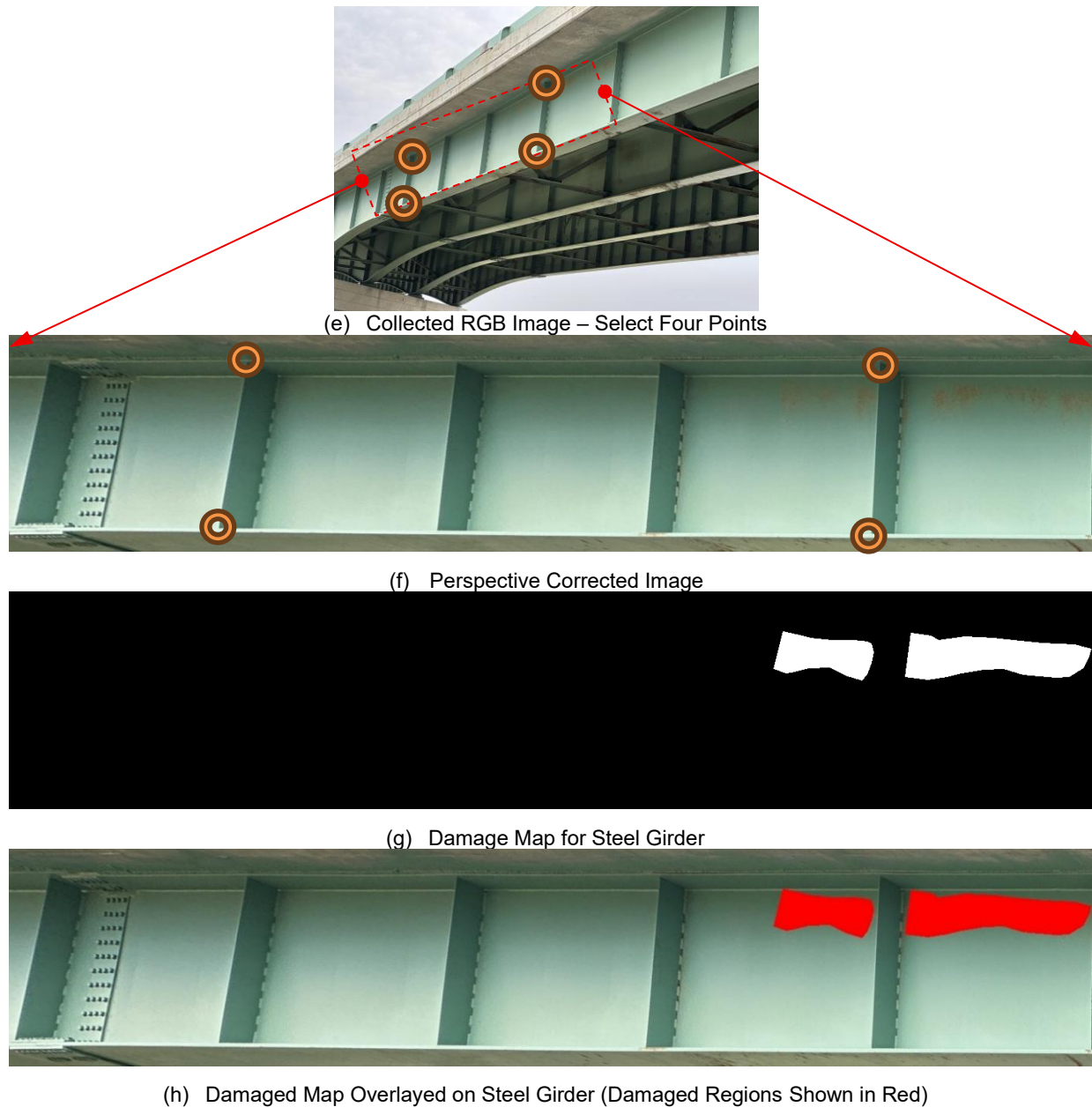


Figure 7.2 – Sample 1 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method

The proposed approach was implemented on different girders and images. For example, in **Fig. 7.3**, the damaged region had an area of 9.72 ft² (0.9 m²). The damaged region of the steel girder shown in **Fig. 7.4** had an area of 5.79 ft² (0.549 m²).



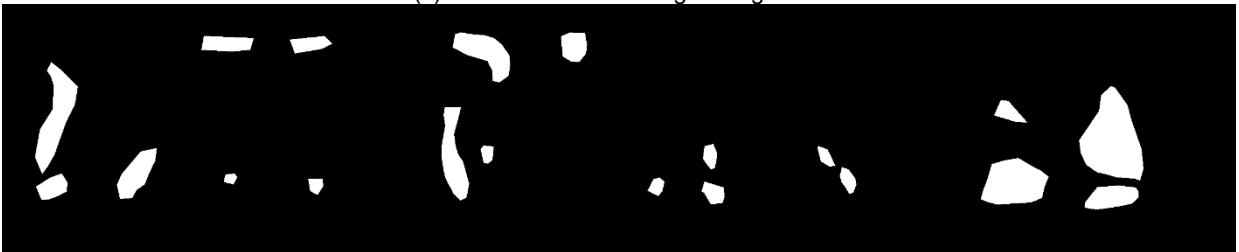
(a) Collected RGB Image



(b) Perspective Corrected Image of Third Girder



(c) Annotation of Damaged Regions



(d) Damage Map for Steel Girder



(e) Damaged Map Overlaid on Steel Girder (Damaged Regions Shown in Red)

Figure 7.3 – Sample 2 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method



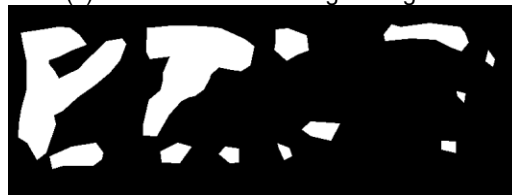
(a) Collected RGB Image



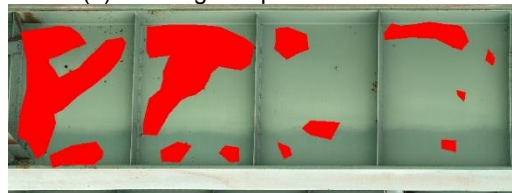
(b) Perspective Corrected Image



(c) Annotation of Damaged Regions



(d) Damage Map for Steel Girder



(e) Damaged Map Overlaid on Steel Girder (Damaged Regions Shown in Red)

Figure 7.4 – Sample 3 of Damage Assessment for Steel Girders Using a Human-in-the-Loop Method

CHAPTER 8 – COMPUTER AIDED BRIDGE INSPECTION AND REPORTING SOFTWARE

8.1 Introduction

In the present study, the research team proposed three methods to detect defects of concrete bridge decks and to generate inspection reports using lidar data, drone based RGB imaging, and drone based thermal imaging. Furthermore, a method to estimate steel girder defects was also developed. In all these methods, computer vision techniques, a field of artificial intelligence (AI), were used.

RGB data analysis for bridge decks using either lidar or drone imaging was discussed in **Ch. 5**. Thermal data analysis for bridge decks was discussed in detail in **Ch. 6**. Further, steel girder defect analysis from RGB images was discussed in **Ch. 7**.

In the lidar based method, a three-dimensional (3D) scan of bridge decks is obtained using iPhone/iPad lidar sensor and augmented reality technique, and then a two-dimensional (2D) image of bridge decks is extracted from the 3D scan (point cloud). Subsequently, a neural network detects and quantifies paint marks from chain-drag inspections. Finally, a delamination map is generated.

In the drone RGB-based method, a 2D map of bridge decks is first obtained by stitching several RGB images and then a neural network detects and quantifies the inspection paint marks. A delamination map is then generated including condition state per defect.

The two RGB methods discussed above rely on the chain-drag method. Nevertheless, a drone thermal-based method can be used to assess the condition of bridge decks without any additional field work such as chain dragging. In days, delaminated regions of concrete appear hotter than solid (undamaged) regions in thermal images. In the proposed method, a 2D thermal map of bridge decks is first generated by stitching several thermal images of decks. Subsequently, the thermal map is analyzed using a neural network to detect the delaminated regions in bridge decks. Finally, a delamination map is generated.

A human-in-the-loop method was proposed for condition assessments of steel girders, in which RGB images from either drones or conventional cameras are used. In this method, RGB images are first corrected using camera parameters, then the perspective view is corrected to show the girders in their elevation view (perpendicular to the girder's web). The perspective correction and scale of image (pixel to length) are provided by the user; thus, it is a human-in-the-loop method. Subsequently, the inspector/user annotates the defected regions of the steel girders. Finally, a defect map is generated.

A user-friendly software, Bridge Computer Vision Inspection Tools (Br.CVIT), was developed by the research team implementing all of the above-mentioned methods and approaches. The software (a screenshot is shown in **Fig. 8.1**) includes five modules to perform: (1) assessment of bridge decks using lidar data, (2) assessment of bridge decks using drone's RGB images, (3) assessment of bridge decks using drone's RGB-thermal images, (4) assessment of steel girders using single RGB image, and (5) calibration of RGB and thermal cameras. A comprehensive

user guide was developed providing step-by-step procedures to perform each assessment. This chapter presents a brief overview of the software.

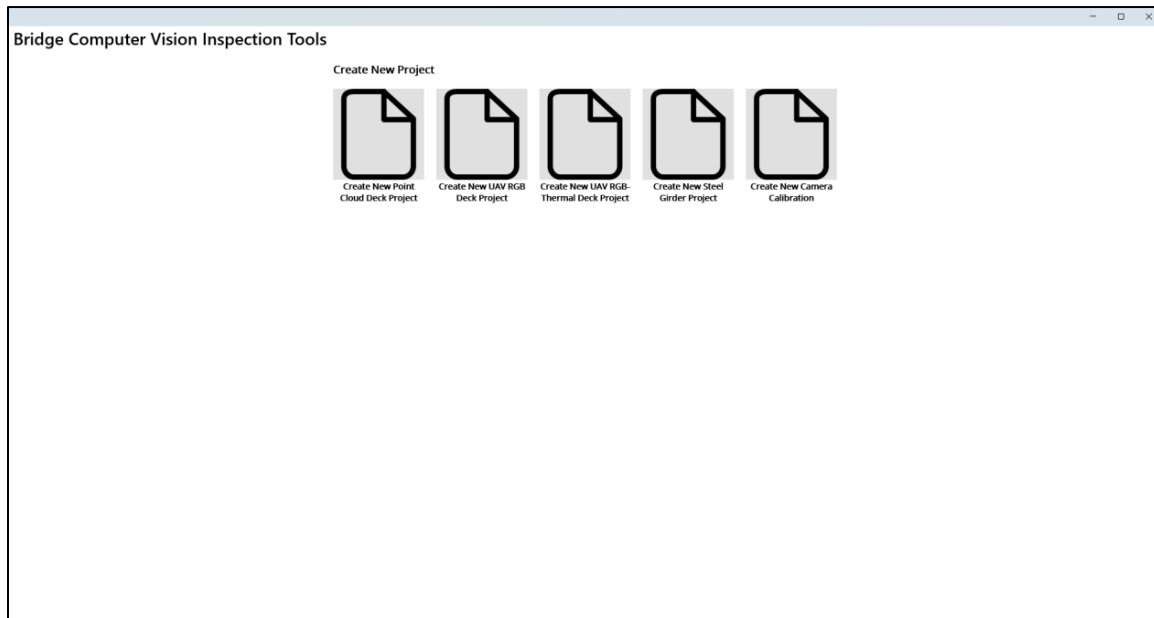


Figure 8.1 – Bridge Computer Vision Inspection Tools (Br.CVIT) Landing Page

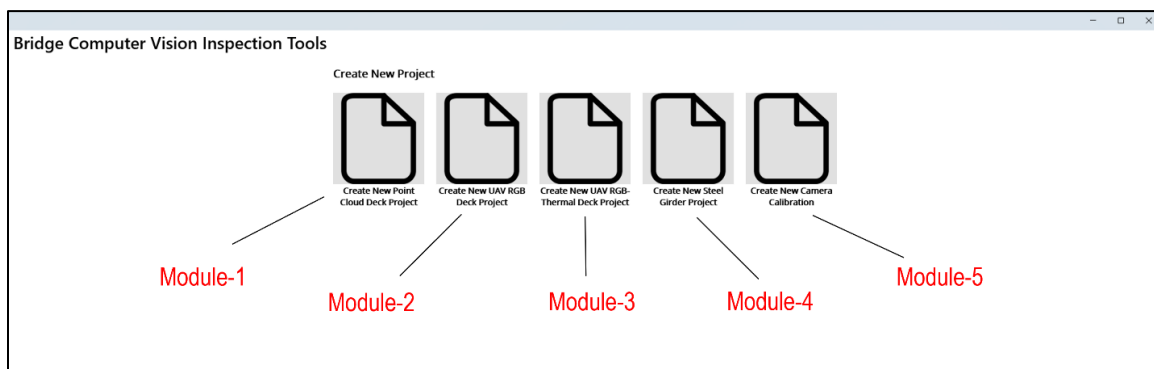


Figure 8.2 – Software Modules

8.2 Software Module for iPhone Based Lidar Inspection of Concrete Bridge Decks

As discussed in **Ch. 5**, the iPhone/iPad lidar imaging can be used to obtain 3D maps of bridge decks after completion of a chain dragging when the deck defects are color marked. After collecting the lidar data, the user can extract the 3D model files using web-based services such as **PolyCam** (**Fig. 8.3**). From PolyCam, export the following files:

- .obj – the 3D object file
- .mtl – the material file
- .jpg – the texture image file

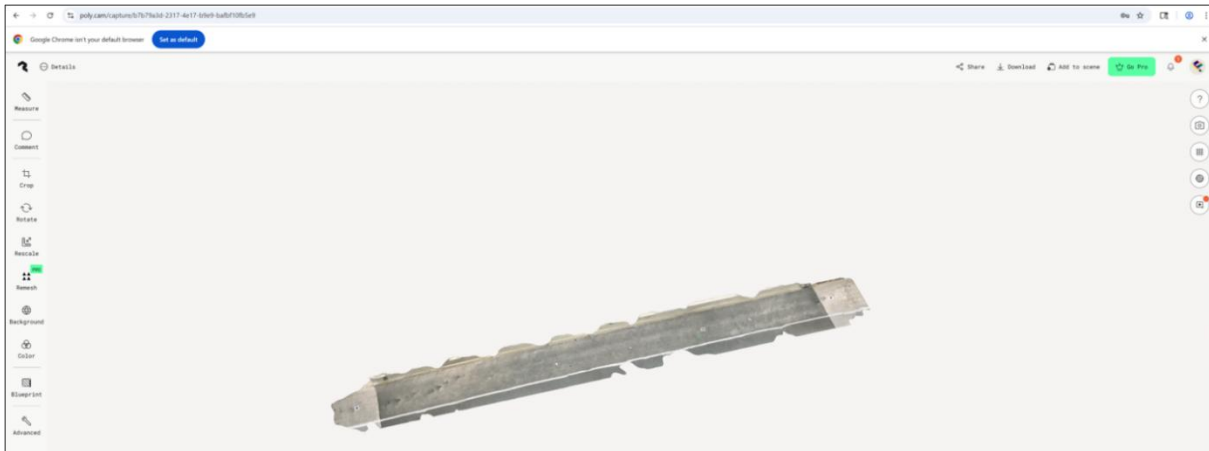


Figure 8.3 – 3D Map of a Bridge Deck Shown in PolyCam

Once the object file has been extracted, the user can use module-1 (**Fig. 8.4**) of the software to generate a damage report.

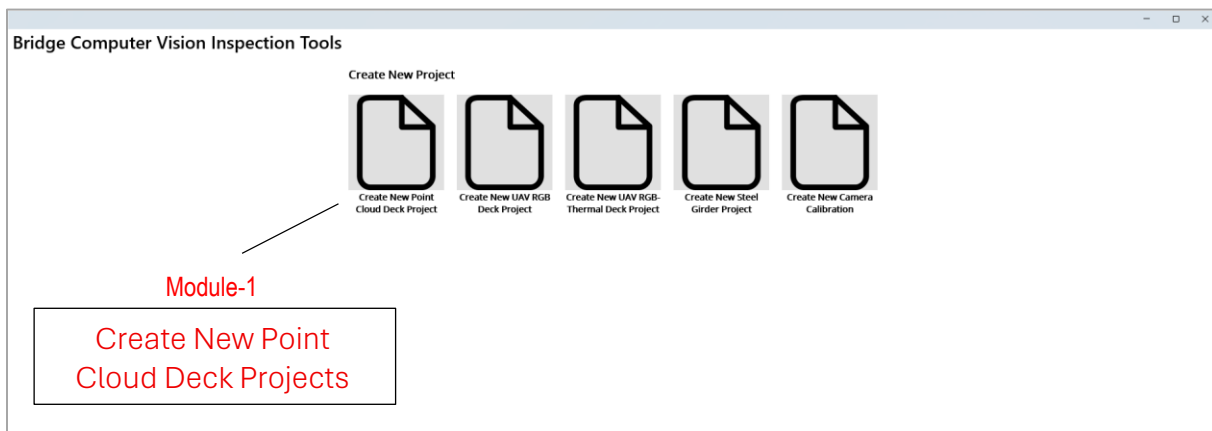


Figure 8.4 – Module-1 for iPhone-based Lidar Data

The software then extracts a 2D map from the 3D mesh of the bridge deck (**Fig. 8.5**).

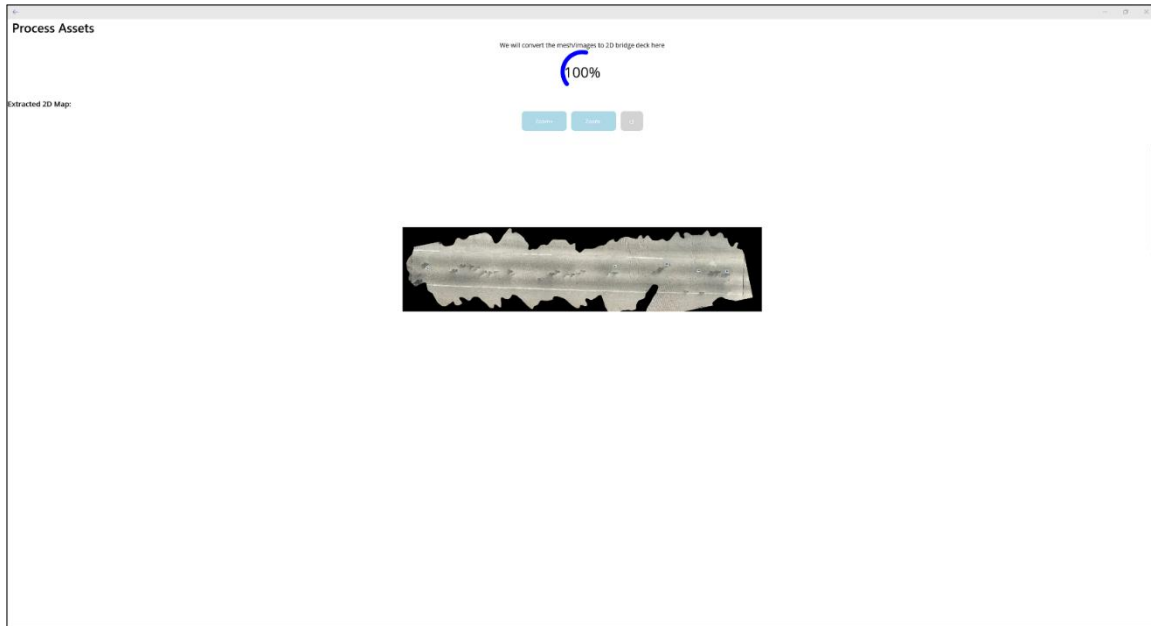
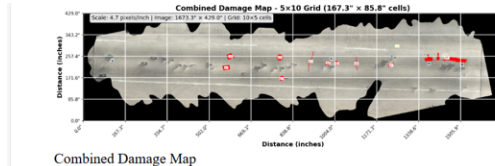


Figure 8.5 – Extracted 2D Map from 3D Mesh using Module-1

The software allows the user to perform either an AI based detection for deck damages or manually mark them. At the end a combined map along with the information related to the damaged regions is generated (**Fig. 8.6**).



Reports:

Delamination Report

Defect ID	Area (px)	Area (in.Å²)	Damage Rating	Grid Cell	Damage Location
d1	4192.5	192.93	4	(7,2)	(1226.8", 222.9")
d2	4030.5	185.48	4	(6,2)	(1010.4", 230.2")
d3	4759.0	219.00	4	(5,2)	(975.0", 230.4")
d4	37789.5	1739.02	4	(8,2)	(1448.4", 245.8")
d5	527.5	24.27	2	(7,3)	(1252.2", 296.0")

Patch Report

Defect ID	Area (px)	Area (in.Å²)	Damage Rating	Grid Cell	Damage Location
p1	7120.0	327.65	4	(4,1)	(790.7", 169.7")
p2	9774.0	449.79	4	(4,2)	(783.9", 252.9")

Spall Report

Defect ID	Area (px)	Area (in.Å²)	Damage Rating	Grid Cell	Damage Location
s1	10450.0	480.89	4	(3,2)	(570.2", 212.6")
s2	10415.5	479.31	4	(3,2)	(585.6", 257.2")

Crack Report

Defect ID	Length (px)	Length (in)	Damage Rating	Grid Cell	Damage Location
c1	384.7	82.52	4	(5,2)	(907.6", 240.8")
c2	259.4	55.65	4	(6,2)	(1089.4", 231.2")

Figure 8.6 – Sample Damage Report Generated by Module-1

8.3 Software Module for Drone Based RGB Inspection of Concrete Bridge Decks

As discussed in **Ch. 5**, the drone-based RGB images can be used to obtain 2D RGB maps of bridge decks including paint markings by inspectors. After collecting the RGB images (**Fig. 8.7** and **8.8**), the user can utilize module-2 of the software to generate a damage report of bridge decks.



Figure 8.7 – Overlapping RGB Images of Bridge Deck with Delamination Marks

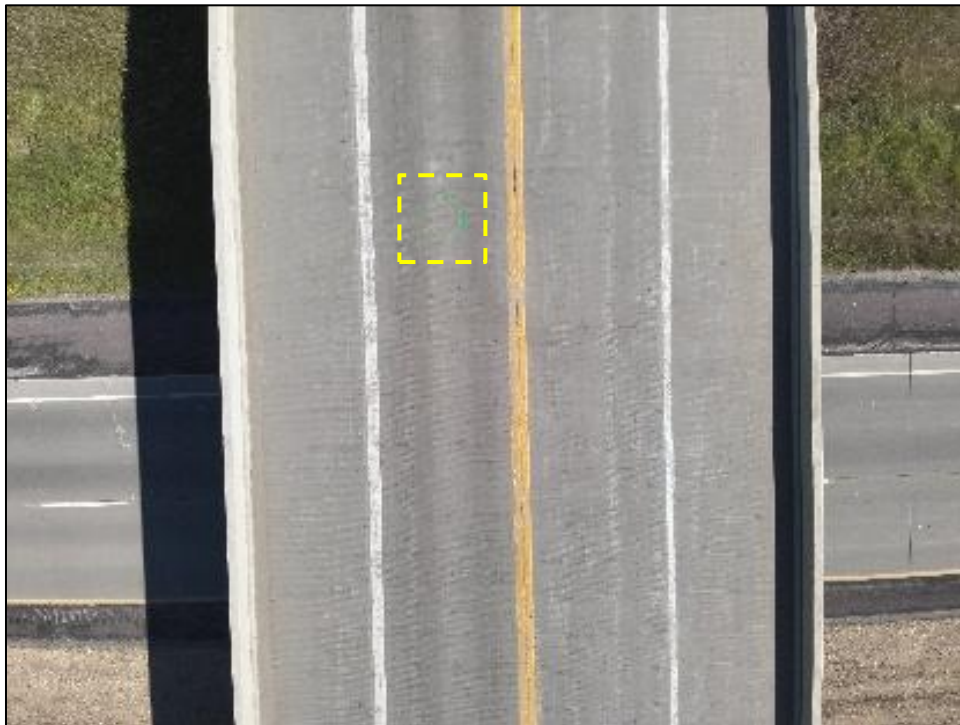


Figure 8.8 – Sample RGB Image of a Bridge Deck with Delamination in Green

Module-2 of the software removes camera distortions and stitches the individual RGB images together to generate a 2D RGB map of bridge decks (**Fig. 8.9**).



Figure 8.9 – Stitched 2D RGB Map of a Bridge Deck

Similar to the previous module, the software allows the user to perform either an AI based detection for deck damages or manually mark them. At the end, a combined damage map along with the information related to the damaged regions is generated (**Fig. 8.10**).

Find Delaminations

We have processed the delamination analysis. You will find detailed analysis below.

Extracted 2D Map:



Detected Defects:



Detected Delaminations



Detected Patched Areas



Detected Spalling



Detected Cracks



Combined Damage Map

Reports:

Delamination Report

Defect ID	Area (px)	Area (in ²)	Damage Rating	Grid Cell	Damage Location
d1	876.5	307.87	4	(4,1)	(2368.3", 439.2")
d2	5695.0	2000.34	4	(7,2)	(4581.8", 447.5")
d3	4954.0	1740.06	4	(5,2)	(3334.3", 475.3")
d4	4801.0	1686.32	4	(6,2)	(4041.3", 605.7")
d5	4488.0	1576.38	4	(3,2)	(1912.5", 612.2")
d6	4730.0	1661.38	4	(4,2)	(2877.9", 612.8")

Patch Report

Defect ID	Area (px)	Area (in ²)	Damage Rating	Grid Cell	Damage Location
p1	3475.0	1220.57	4	(6,2)	(3638.3", 619.3")
p2	3469.5	1218.64	4	(6,2)	(3780.6", 635.3")

Spall Report

Defect ID	Area (px)	Area (in ²)	Damage Rating	Grid Cell	Damage Location
s1	5598.0	1966.26	4	(6,2)	(3640.1", 455.2")
s2	6737.5	2366.51	4	(5,2)	(3372.8", 623.5")

Crack Report

Defect ID	Length (px)	Length (in)	Damage Rating	Grid Cell	Damage Location
c1	145.2	86.06	4	(5,2)	(3090.1", 622.7")
c2	188.5	111.71	4	(5,2)	(3128.3", 454.1")

Figure 8.10 – Sample Damage Report Generated by Module-2

8.4 Software Module for Drone Based Thermal Inspection of Concrete Bridge Decks

As discussed in **Ch. 6**, drone-based thermal imaging can be used to obtain 2D thermal maps of bridge decks which can provide an insight to the condition of bridge decks. The hot spots, which are the potential delaminated regions, can be identified by the neural network. After collecting the RGB and thermal images (**Fig. 8.11** and **8.12**), the user can utilize module-3 of the software to generate a damage report of bridge decks.



Figure 8.11 – Overlapping RGB Images of Bridge Deck

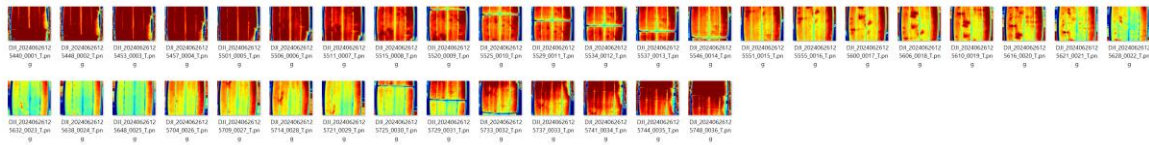


Figure 8.12 – Overlapping Thermal Images of Bridge Deck

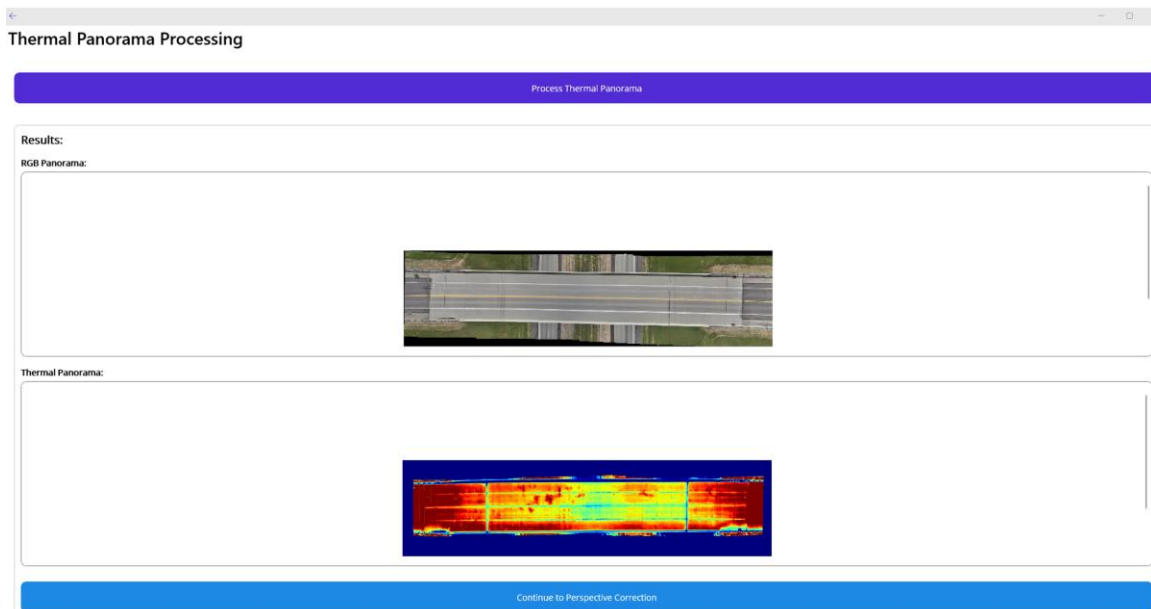


Figure 8.13 – Stitched RGB and Thermal Maps of a Bridge Deck

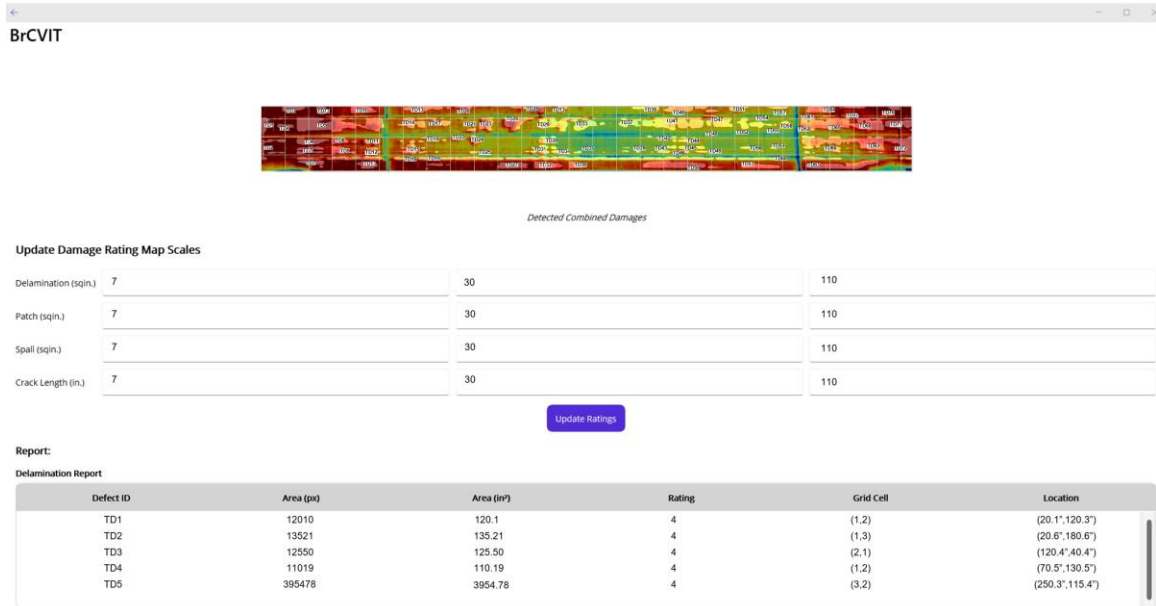


Figure 8.14 – Sample Damage Report Generated by Module-3

8.5 Software Module for Drone Based Inspection of Steel Girders

As discussed in **Ch. 7**, RGB images of steel girders can be used to obtain a defect map. After collecting the RGB images (**Fig. 8.15**), the user can utilize module-4 of the software to generate a damage report for steel girders.

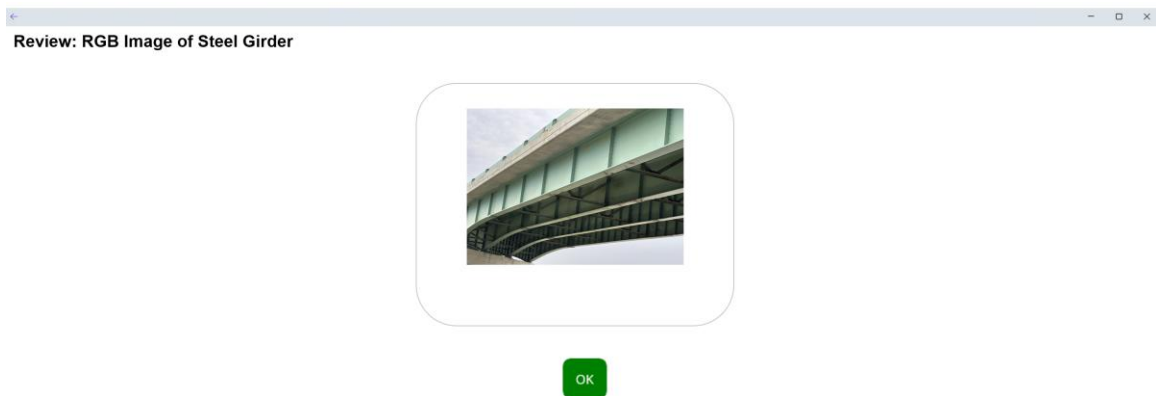


Figure 8.15 – RGB Image of Steel Girders

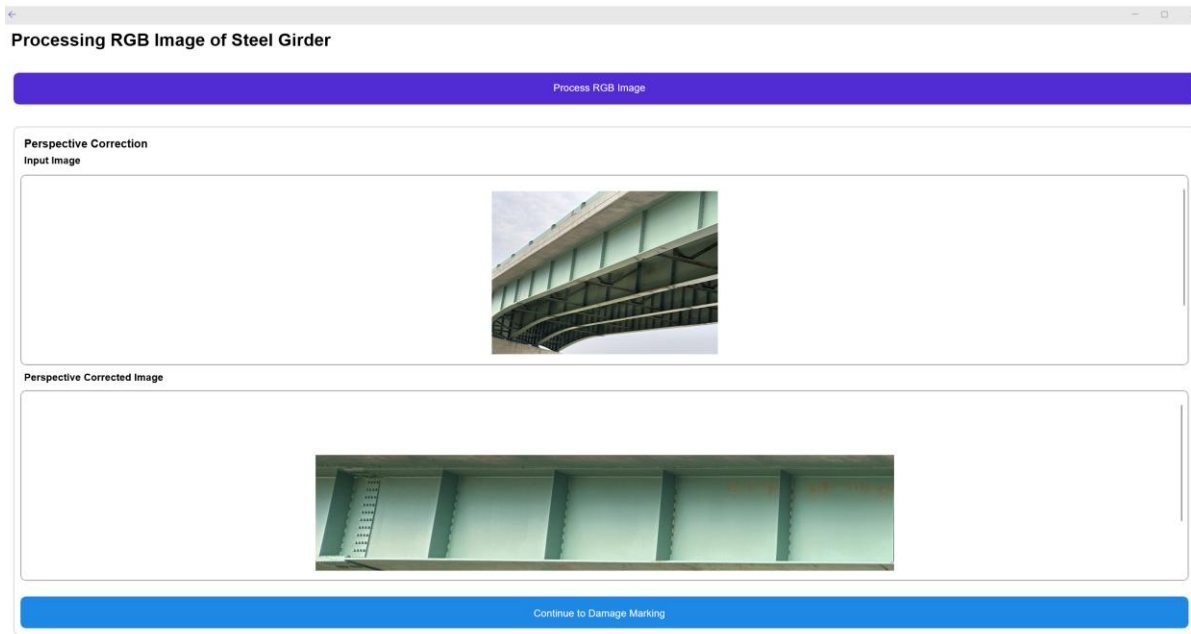


Figure 8.16 – Corrected Perspective of Steel Girder Images Using Four Known Points

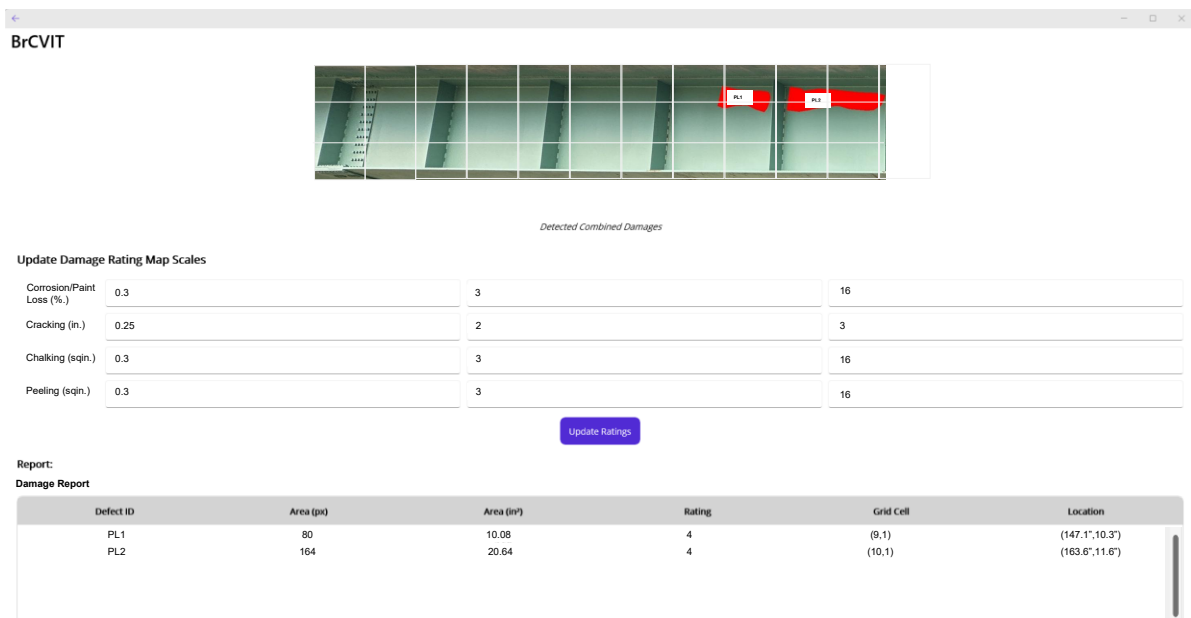


Figure 8.17 – Sample Damage Report Generated by Module-4

8.6 Software Module for Camera Calibration

As discussed in **Ch. 5**, camera calibration parameters are needed to stitch RGB and thermal images. After collecting the RGB and/or thermal images (**Fig. 8.18**), the module-5 of the software can be used to obtain camera calibration parameters.

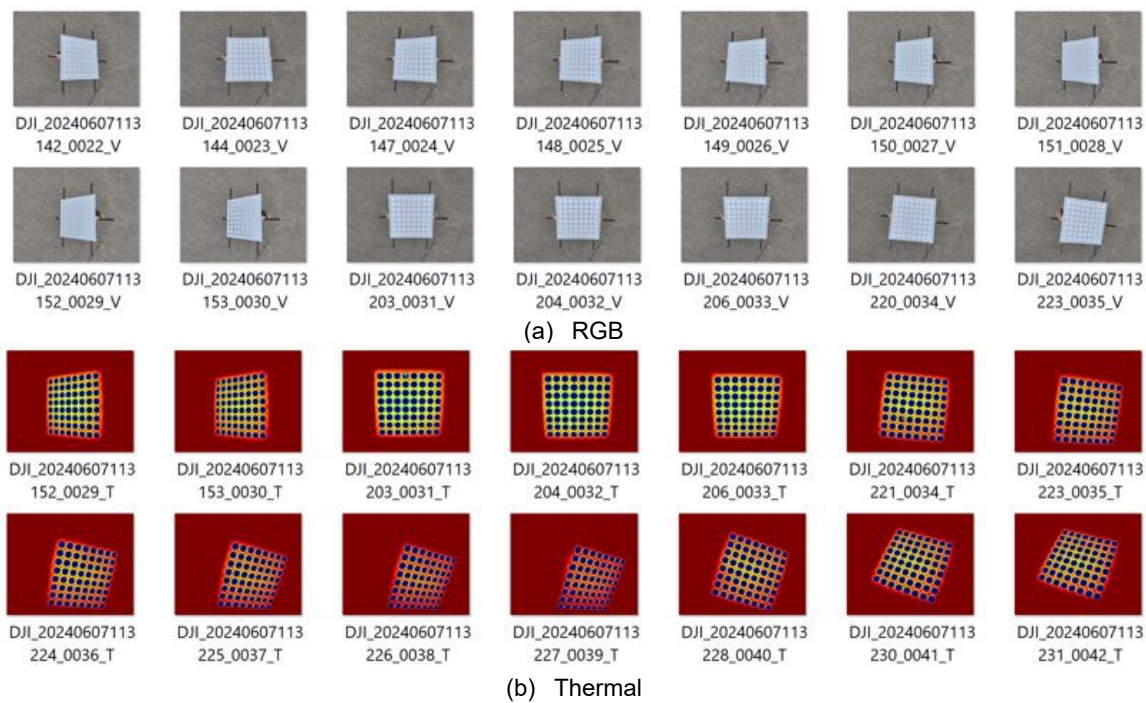


Figure 8.18 – Sample RGB and Thermal Images for Camera Calibration

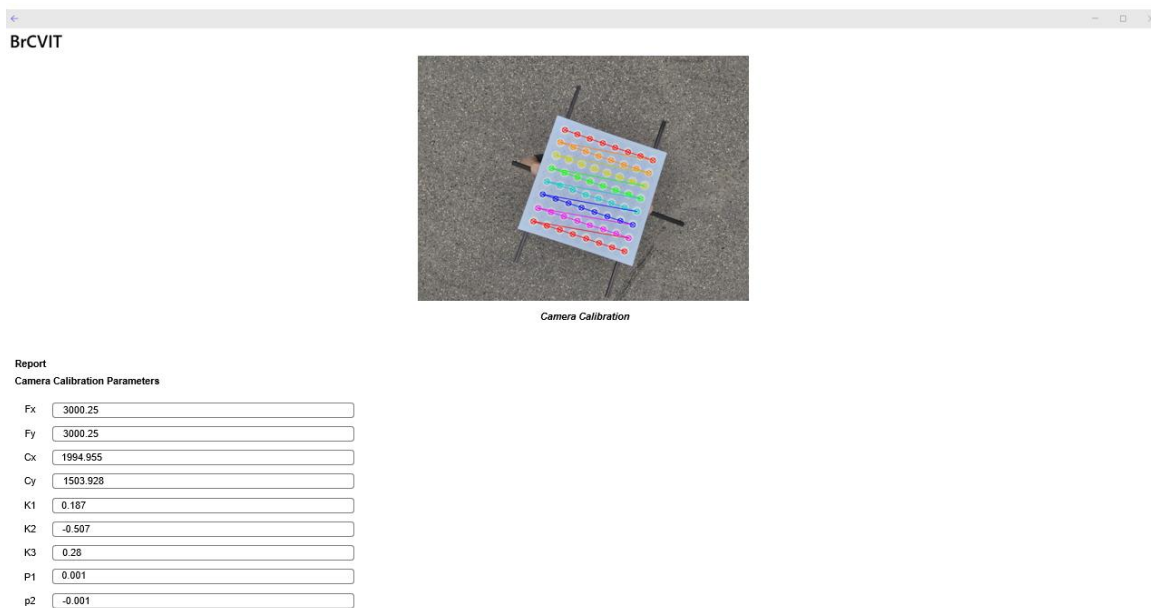


Figure 8.19 – Sample RGB Camera Calibration Parameters Generated by Module-5

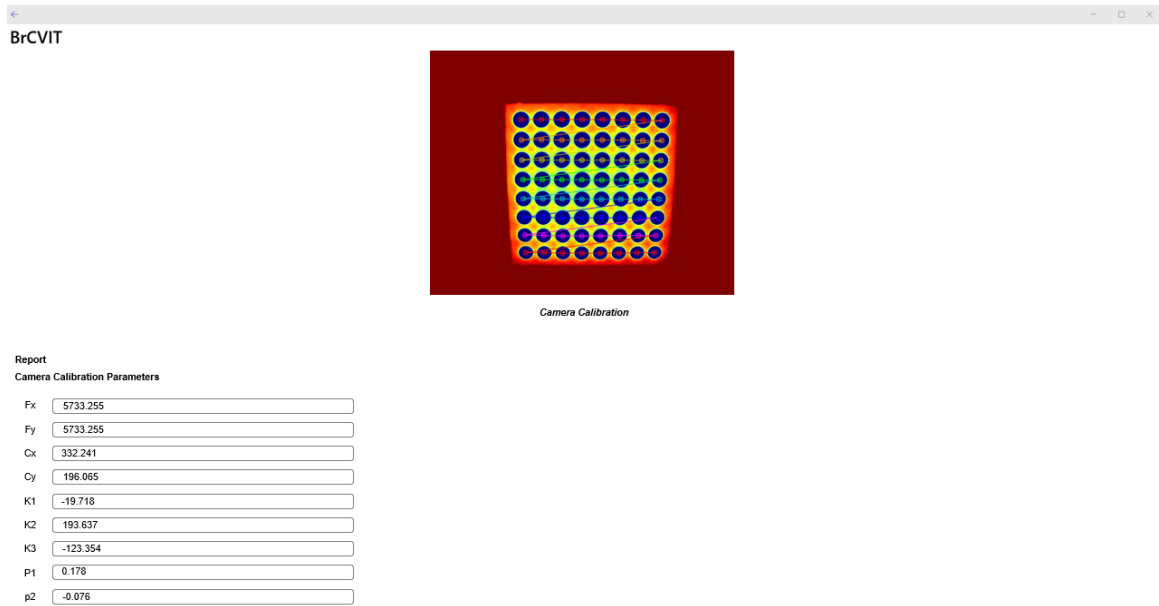


Figure 8.20 – Sample Thermal Camera Calibration Parameters Generated by Module-5

CHAPTER 9 – BEST PRACTICES FOR DATA COLLECTION

9.1 Introduction

During the course of the project, the research team exercised different data collection methods and practices, and found some of those, especially for lidar and thermal data, crucial for successful post-processing by the software. This chapter provides guidelines on the best practices to obtain inspection data including iPhone/iPad lidar imaging, drone RGB imaging, and drone thermal imaging.

9.2 Best Practices to Collect Lidar Data

To obtain iPhone/iPad point-cloud data, PolyCam App, one of many applications available for such data, can be used. The lidar data can then be exported to the software (Br.CVIT) using the Object format. The research team recommends the following as the best practices when collecting the lidar data.

- Scan the bridge deck at height of 5 ft or less.
- Scan one lane at a time.
- Scan along the sunlight to avoid glare.

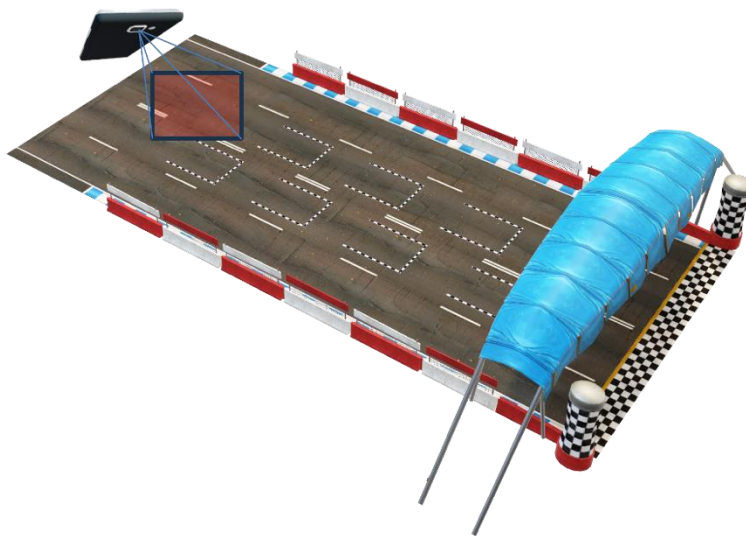


Figure 9.1 – Data Collection Using iPhone/iPad Lidar Sensor

9.3 Best Practices to Collect RGB and Thermal Data

To obtain the RGB and thermal data of bridge decks, the research team recommends the following guidelines for the best outcomes.

9.3.1 Flying Height of Drone from Concrete Bridge Decks

Based on the resolution of RGB and thermal sensors, the drone flying height must be adjusted for preferred dimensions that can cover a lane, multiple lanes, or the full width of the bridge. The field of view for thermal cameras is narrower than that of RGB cameras, therefore it usually

controls the flying altitude for a desired view.

For example, and as shown in **Fig. 9.3**, the entire width and length of a bridge can be viewed in a single image captured at a height of 300 ft (91 m) using a DJI Mavic 3T drone. However, the resolution of such image is low for AI models. Multiple images of the bridge deck can be collected at a lower height to be stitched together to obtain high resolution thermal map of the entire bridge deck. In such cases, the flying height of the drone should be adjusted to obtain the full width of the bridge deck. For example (**Fig. 9.3**), flying at a height of 15 ft (4.6 m) from the bridge deck covers a single lane of the bridge. To cover the full width of the bridge in **Fig. 9.3**, including two lanes each 12 ft (3.65 m) wide and two shoulders each 10 ft (3 m) wide, the drone was at a 50 ft (15 m) high from the bridge deck.

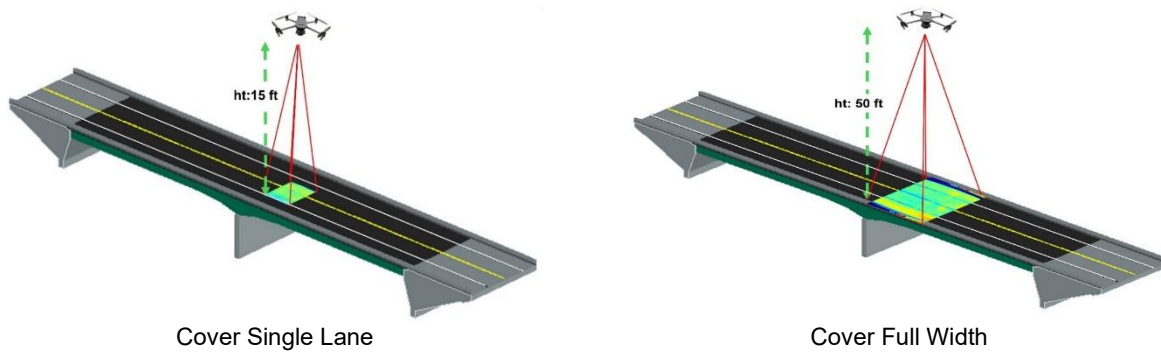


Figure 9.2 – Flying Height of Drone to Cover Single Lane and Full Width of Bridges

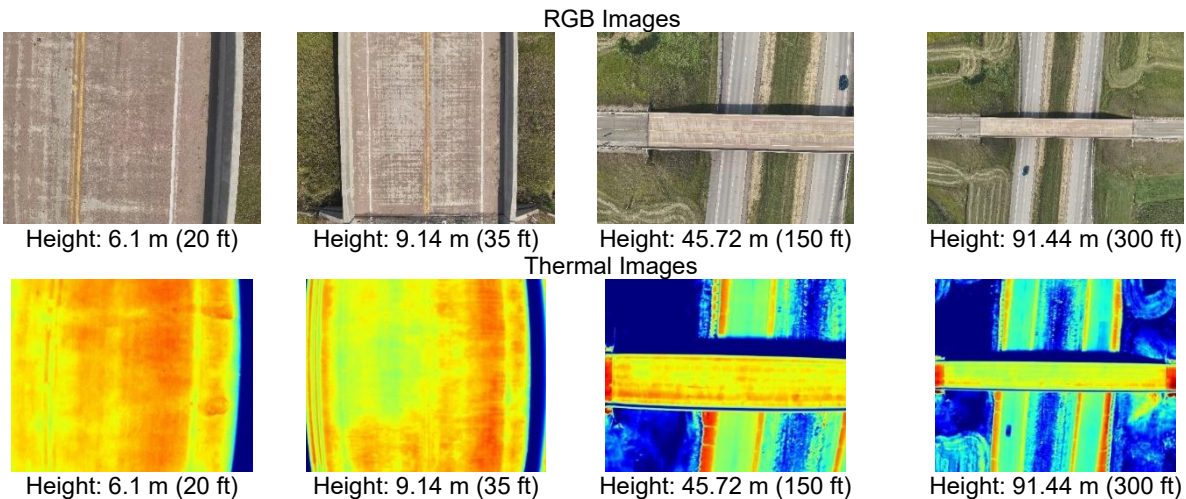
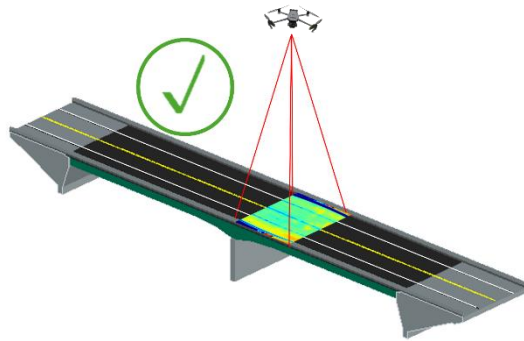


Figure 9.3 – Sample of Drone RGB-Thermal Images Collected at Different Heights from Bridge Decks

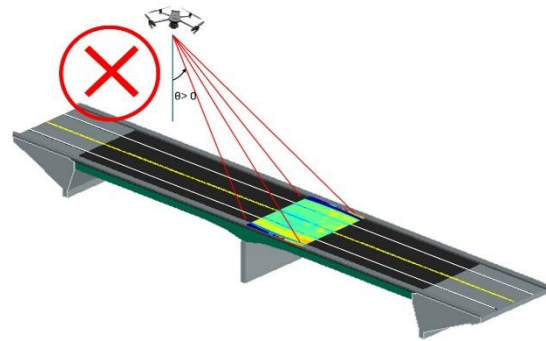
9.3.2 Gimble Angle

To better stitch individual images after collecting multiple images of a bridge deck, it is important to keep the gimble pointed directly downward (Nadir) instead of having a slanting angle (**Fig. 9.4**). **Figure 9.5** provides sample images captured with different gimble angles. The images collected at a slanting angle cause problems in image stitching procedure. For example, **Fig. 9.6** compares stitched image of a bridge based on the data collected at a slanting angle with

a stitched image obtained from images collected at the recommended angle.

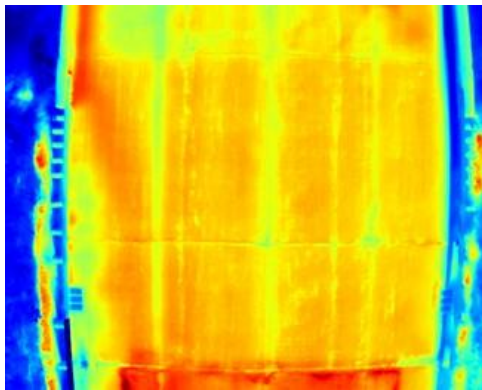


Nadir - Point Directly Downward (Recommended)

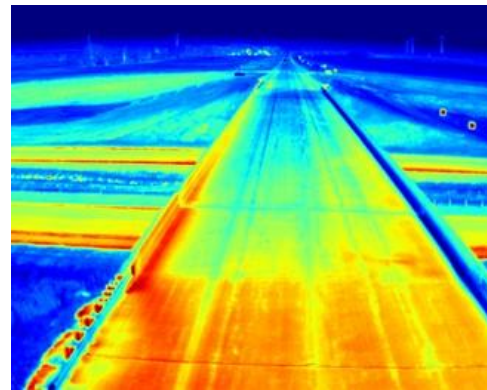


Slanting Angle (Not Recommended)

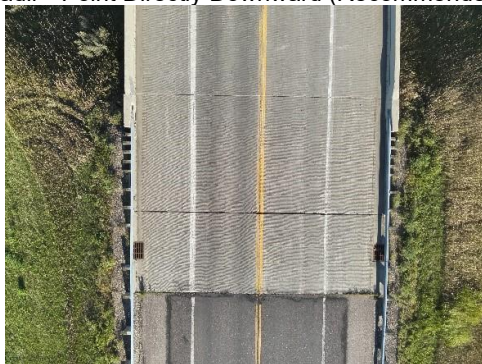
Figure 9.4 – Recommended Gimble Angle for RGB and Thermal Imaging



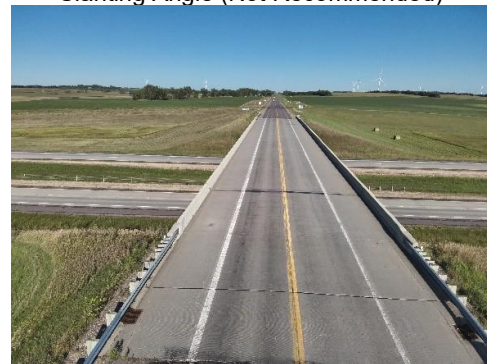
Nadir - Point Directly Downward (Recommended)



Slanting Angle (Not Recommended)



Nadir - Point Directly Downward (Recommended)



Slanting Angle (Not Recommended)

Figure 9.5 – Sample Images with Different Gimble Angles

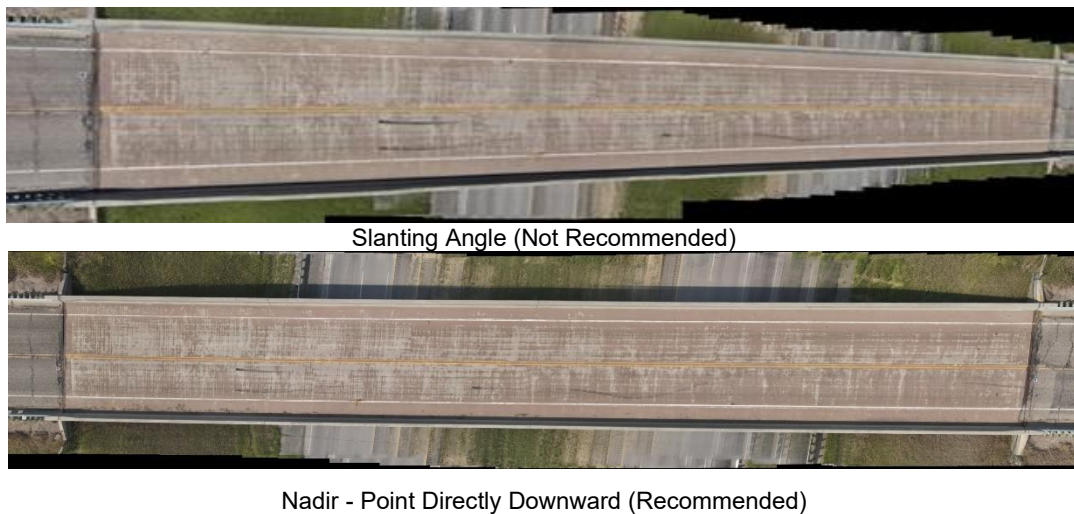


Figure 9.6 – Comparison of Stitched Images Based on Data Collected at Slanting Angle and Pointing Downward

9.3.3 Overlap Between Thermal Images

There should be an overlap between the successive RGB or thermal images of bridge decks. An overlap of 30 to 60% is recommended (**Fig 9.7**). Note that since the field of view of existing thermal cameras is narrower than that of RGB, more images might be needed when collecting thermal data. Adequate overlap provides a better image stitching outcome for bridge decks.

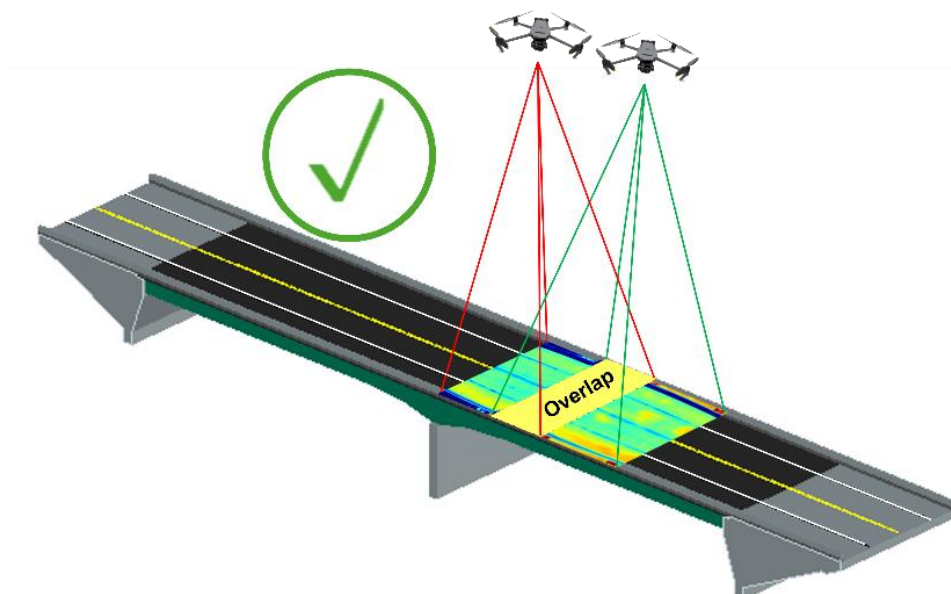


Figure 9.7 – Required Overlap in Successive Drone Imaging

9.3.4 Suitable Time Period to Collect Thermal Image Dataset from Bridge Sites

Bridge inspections usually start early in the morning and last for the whole day. However, thermal gradients of the bridge decks may vary substantially from sunrise to sunset. It is crucial to find the best times of days in which thermal data is viable and sufficient to detect deamination. Note one may collect thermal data at night; however, it is not common, and the neural network trained in the present project works only on the day data, where delaminated areas of concrete are hotter.

To understand the best timeframe for thermal data collection, thermal images were captured for a bridge from 6 a.m. to 8 p.m. with two-hour intervals during a summer day at a latitude of 45.1078764°N (South Dakota, USA). **Figure 9.8** shows the stitched thermal maps of the bridge deck for these 14 hours. At 6 a.m., the delaminated regions were colder as compared to the intact regions of the concrete deck. As the bridge absorbs heat during the daytime, the delaminated regions heat up more than the intact regions thus they appear as hot spots in the images. From noon to 4 p.m., the delaminated regions were conspicuous as easily seen in the figures as hot spots. In the evening, as the bridge got cooler, the delaminated regions started to fade out.

To explore the deep learning model performance on these 14-hour data collection scheme, the central portion of the stitched images, the two traffic lanes, were cropped and then fed to the neural network. **Figure 9.9** shows the model detected delaminated regions. Using the morning data, the neural network was not able to detect correct delaminated regions as the temperature of these regions was lower than the intact regions. However, the model was consistent in detecting delamination using data obtained between noon and 4 p.m.

It is understood that location and weather also play a significant role in collecting suitable thermal data for bridges. Nevertheless, as a general guide and based on thermal maps and deep learning model performance to detect delamination, the study proposes to collect thermal images from concrete bridge decks from 11 a.m. to 5 p.m. at latitudes less than 46°N in the northern hemisphere. The time window is expected to be shorter in higher latitudes such as Alaska.

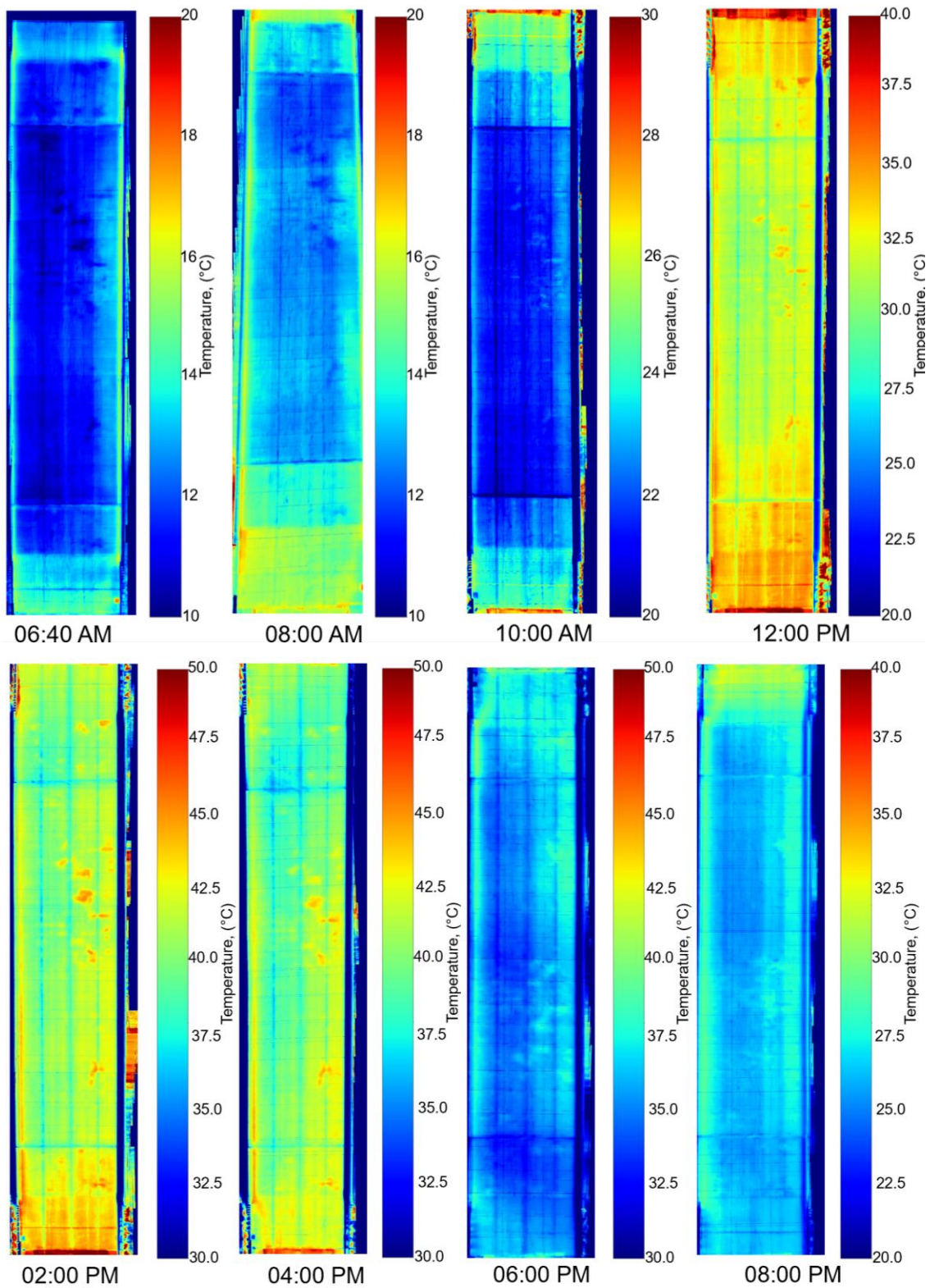


Figure 9.8 – Suitable Time for Thermal Data Collection

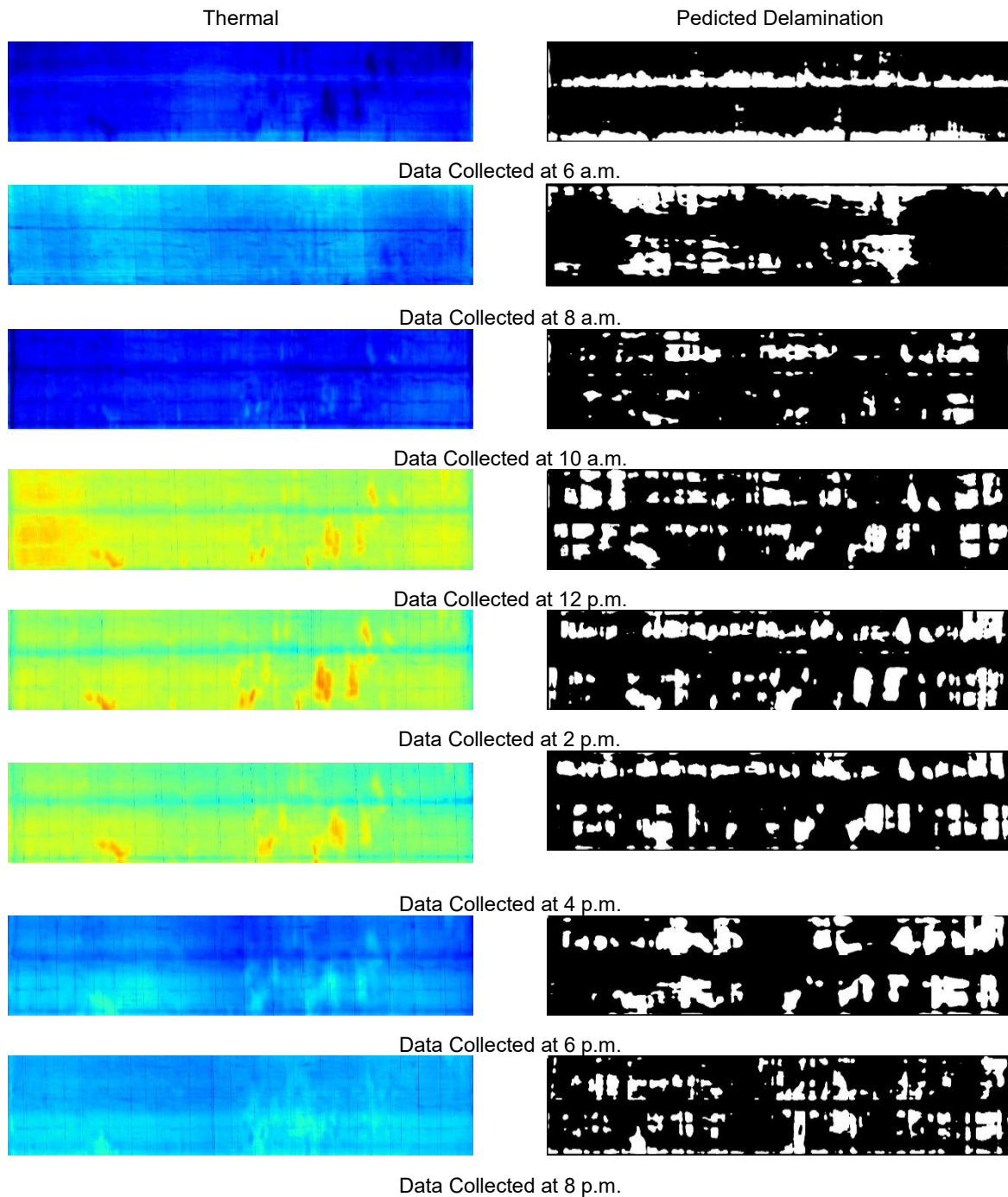


Figure 9.9 – Delamination Detection by Neural Network for Thermal Images Collected at Different Time of a Summer Day

CHAPTER 10 – SUMMARY AND CONCLUSIONS

10.1 Summary

Regular maintenance of transportation infrastructure such as bridges is crucial in extending their service life while reducing long term repair costs. Transportation agencies such as the Federal Highway Administration (FHWA) seek emerging technologies and innovative programs to enhance asset management and maintenance. Furthermore, timely inspections of bridges by trained personnel are mandated per national and state specifications. The Alaska Department of Transportation and Public Facilities (DOT&PF) engineers inspect approximately 500 bridges each year. In Alaska, approximately 44% of bridges are in good condition, 49% are in fair condition, and 7% are rated poor based on the 2023 National Bridge Inventory (NBI) data.

In general, visual inspection is the common practice for inventory and routine inspections and is combined with other tools such as nondestructive evaluation (NDE) in other inspections for enhanced assessment. The inspector must complete both an NBI inspection and an element level inspection for each bridge. Nationwide, most steps of bridge inspections such as data collection, defect quantification, and damage reporting are done manually. Furthermore, each of these steps is time consuming, error prone, and hard to repeat in the following inspections due to the manual nature of these activities. Such measurements often require traffic control for the safety of the inspection crew, and the findings might not be consistent if done by different inspectors.

Computer vision, a field of artificial intelligence (AI) that can analyze scenes at human level performance, can speed up defect identification and quantification only using images of bridge elements. Progress in computer aided analysis of visual imagery has initiated widespread use of computer vision tools in various tasks, including but not limited to object classification, categorization, detection, and generation by learning from provided datasets for specific tasks. Even though the academic and industry level developments are interesting and on the right track for field deployment, they usually lack practicality and are overwhelming for everyday users. For example, the academic-level computer programs are not as user-friendly as commercial software, and the industry-level software packages are so enlarged targeting multiple assets preventing them from being practical. In summary, current tools do not fully address the needs of DOTs for routine bridge inspections.

The primary objective of this project was to develop practical AI tools that help inspectors with defect detection and measurement and facilitate the inspection reporting following standard practices. To achieve the project goals, a comprehensive literature review of the current scientific trend and AI tools was performed on delamination detection, bridge map generation, and damage state estimation. Then, a new definition of condition states for delamination of concrete bridge decks and steel girder defects suitable for computer programming was proposed. Subsequently, more than 50 bridges were inspected, and a comprehensive dataset of inspector annotated delamination was compiled using iPhone-based lidar, conventional cameras, and drone based RGB and thermal cameras. Images of steel girders with and without damages were also collected. A novel approach to generate a thermal map for entire bridge using RGB and thermal images from drones was developed. Calibration of RGB and thermal cameras was carried out to undistort the images. Homography matrices obtained from RGB image stitching were utilized to generate thermal maps of the entire bridge decks. Best practices to collect lidar, RGB and

thermal data from concrete bridge decks were also proposed. Finally, an AI-based tool that can detect, quantify, and map the inspector's annotations, and produce a delamination map of whole bridge deck including their condition states was developed for RGB images. A separate AI-based tool was developed for thermal images to identify the potential delaminated regions. Since data normalization usually impacts training and model accuracy, seven different normalization methods based on the mean, standard deviation, minimum and maximum temperatures for thermal images were evaluated to obtain the best prediction of delamination. Furthermore, for inspecting bridge steel girders, a simple tool with a human-in-the-loop method was developed to allow users to remove camera distortions from RGB images, annotate steel girder defects such as paint loss in the image, and obtain the physical area of the damage. A user-friendly software, Bridge Computer Vision Inspection Tools (Br.CVIT), was developed by the research team implementing all of the above-mentioned methods and approaches.

10.2 Conclusions

The following conclusions can be drawn from this computer vision-based study on bridge inspections:

- A new condition state definition was developed for delamination of concrete bridge decks. The new definition enhances the physical interpretation of existing criteria in AASHTO by using definitive and quantitative measures at all four levels that are suitable for computer programming and automated analysis. The new definition ensures greater alignment between manual assessments and computer-based analyses, improving the accuracy and consistency of damage evaluations. The definitions were implemented in Br.CVIT as default values and was found viable and repetitive.
- A similar definition was developed for steel girders and were implemented in Br.CVIT.
- A new image-lidar dataset of inspector annotated delamination was compiled after inspecting more than 50 bridges. The images are particularly applicable in training neural network models to automatically detect and measure inspector-annotated delaminated areas. This dataset contains 1,918 RGB photographs of inspector annotations and 18 lidar scans of bridge decks.
- For detections of the inspector annotations on bridge decks in RGB images, three neural network models for image segmentation (U-Net, Attention U-Net, and Recurrent Residual U-Net) were evaluated following hyper-parameter tuning, image preprocessing, and augmentation. The color and geometric augmentations of a somewhat small training dataset expanded the dataset to 70,000 to improve generalization and accuracy. U-Net was found to be the best model.
- To detect delamination in thermal images, six neural networks for image segmentation (U-Net, Attention U-Net, Residual U-Net, Transformer ResU-Net, Recurrent Residual U-Net, and ResUNet++) were evaluated. U-Net was found to be the best model.
- Among seven methods to normalize the thermal image dataset, the normalization method based on the individual mean temperature and standard deviation performed better.
- A method to generate high-quality bridge deck maps was developed to extract 2D bridge deck maps from reconstructed 3D mesh (point cloud) data. This allows high accuracy in quantification of the annotated area by inspectors.
- Thermal maps for several bridges were successfully generated using a novel approach based on RGB and thermal image stitching.

- A two-level assessment of the detection of delaminated areas in bridge decks using lidar data was conducted. Local delamination assessment, which evaluates pixel-to-pixel correspondences, was conducted to measure the quality of the granular prediction. The pixel-to-pixel level accuracy for RGB image was less than 75% for the best model. The global delamination assessment (area-to-area) exhibited, on the other hand, a near-to-perfect performance with 99.5% accuracy over 18 bridges. Furthermore, the global assessment confirmed that, regardless of the quality of the local assessment, the framework can quantify the area inside the delaminated spot with high accuracy. The accuracy of the neural network in detecting delaminated areas of 2D RGB maps of bridge decks obtained from drone RGB images was at 96%.
- For thermal image dataset, the best U-Net model showed an accuracy of 77% to 83% in terms of intersection over union to predict delamination in concrete bridge decks. As a global performance assessment, it was found that the total physical delaminated area of bridge decks detected by the AI model was on average 96% that of the ground truth for seven concrete bridge decks, indicating that the model is very accurate.
- The proposed defect detection method for steel girders was also simple and reasonably accurate.

Overall, the proposed multi-device computer-vision based study can expediate bridge deck inspections for delamination by automating the entire process of data analysis for point clouds, RGB, and thermal images, and damage identification and condition state estimation using neural networks.

REFERENCES

- AASHTO MBEL. (2019). “Manual for Bridge Element Inspection, 2nd Edition, with 2022 Interim Revisions,” American Association of State Highway and Transportation Officials (AASHTO), 126 pp.
- Alaska BSM. (2023). “Alaska Bridges and Structures Manual,” Alaska Department of Transportation and Public Facilities, Juneau, AK, 386 pp.
https://dot.alaska.gov/stwddes/desbridge/assets/pdf/manual/bridge_manual.pdf
- Alcantarilla, P. F., & Solutions, T. (2011). “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces,” *IEEE Trans. Patt. Anal. Mach. Intell.*, 34(7), 1281-1298.
<http://dx.doi.org/10.5244/C.27.13>
- Ali, R., Kang, D., Suh, G., & Cha, Y. J. (2021). “Real-time Multiple Damage Mapping Using Autonomous UAV and Deep Faster Region-based Neural Networks for GPS-denied Structures,” *Automation in Construction*, 130, 103831. <https://doi.org/10.1016/j.autcon.2021.103831>
- Aljagoub, D., Na, R., Cheng, C., & Shen, Z. (2022). “Performance Evaluation of Uncooled UAV Infrared Camera in Detecting Concrete Delamination,” *Infrastructures*, 7(12), 163.
<https://doi.org/10.3390/infrastructures7120163>
- Alom, M. Z., Hasan, M., Yakopcic, C., Taha, T. M., & Asari, V. K. (2018). “Recurrent Residual Convolutional Neural Network Based on U-Net (R2U-Net) for Medical Image Segmentation,” *arXiv preprint arXiv:1802.06955*. <https://doi.org/10.48550/arXiv.1802.06955>
- Arya, D., Maeda, H., Ghosh, S. K., Toshniwal, D., & Sekimoto, Y. (2022). “RDD2022: A Multi-national Image Dataset for Automatic Road Damage Detection,” *Geoscience Data Journal*, 11(4), 846-862. <https://doi.org/10.6084/m9.figshare.21431547.v1>
- ASTM C876-22b. (2022). “Standard Test Method for Corrosion Potentials of Uncoated Reinforcing Steel in Concrete,” American Society for Testing and Materials (ASTM) International: West Conshohocken, PA, USA. DOI: 10.1520/C0876-22B
- Bay, H., Tuytelaars, T., & Van Gool, L. (2006, May). “Surf: Speeded up Robust Features,” In *European conference on computer vision* (pp. 404-417). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/11744023_32
- Bowman, M. D., Hagan, B. D., & Hurdle, W. D. (2022). “Steel Bridge Coating Evaluation and Rating Criteria,” *Joint Transportation Research Program Publication No. FHWA/IN/JTRP-2022/23*. West Lafayette, IN: Purdue University. <https://doi.org/10.5703/1288284317386>
- Bradski, G. (2014). “The OpenCV Reference Manual,” <http://docs.opencv.org/>
- Bianchi, E., Abbott, A. L., Tokekar, P., & Hebdon, M. (2021). “COCO-bridge: Structural Detail Data Set for Bridge Inspections,” *Journal of Computing in Civil Engineering*, 35(3), 04021003.

[https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000949](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000949)

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). "BRIEF: Binary Robust Independent Elementary Features," In *Computer Vision—ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5–11, 2010. Proceedings, Part IV* (pp. 778–792). Springer. https://doi.org/10.1007/978-3-642-15561-1_56

Cantzler, H. (1981). "Random Sample Consensus (RANSAC)," *Institute for Perception, Action and Behaviour, Division of Informatics, University of Edinburgh*, 3, 1–4. <https://www.cvl.iis.u-tokyo.ac.jp/~oishi/Papers/Alignment/ransac.pdf>

Charron, N., McLaughlin, E., Phillips, S., Goorts, K., Narasimhan, S., & Waslander, S. L. (2019). "Automated Bridge Inspection Using Mobile Ground Robotics," *Journal of Structural Engineering*, 145(11), 04019137. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0002404](https://doi.org/10.1061/(ASCE)ST.1943-541X.0002404)

Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). "Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE transactions on pattern analysis and machine intelligence*, 40(4), 834–848. <https://doi.org/10.1109/TPAMI.2017.2699184>

Cheng, C., Shang, Z., & Shen, Z. (2020). "Automatic Delamination Segmentation for Bridge Deck Based on Encoder-decoder Deep Learning through UAV-based Thermography," *NDT & E International*, 116, 102341. <https://doi.org/10.1016/j.ndteint.2020.102341>

Cheng, C., & Shen, Z. (2020). "The Application of Gray-scale Level-set Method in Segmentation of Concrete Deck Delamination Using Infrared Images," *Construction and Building Materials*, 240, 117974. <https://doi.org/10.1016/j.conbuildmat.2019.117974>

Dalal, N., & Triggs, B. (2005). "Histograms of Oriented Gradients for Human Detection," In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1* (pp. 886–893). IEEE. <https://doi.org/10.1109/CVPR.2005.177>

Dinh, K., Gucunski, N., Kim, J. & Duong, T.H. (2016). "Understanding Depth-amplitude Effects in Assessment of GPR Data from Concrete Bridge Decks," *NDT & E International*, 83, pp.48–58. <https://doi.org/10.1016/j.ndteint.2016.06.004>

Dong, C. Z., & Catbas, F. N. (2021). "A Review of Computer Vision-based Structural Health Monitoring at Local and Global Levels," *Structural Health Monitoring*, 20(2), 692–743. <https://doi.org/10.1177/1475921720935585>

Dorafshan, S., Thomas, R. J., & Maguire, M. (2018). "SDNET2018: An Annotated Image Dataset for Non-contact Concrete Crack Detection Using Deep Convolutional Neural Networks," *Data in brief*, 21, 1664–1668. <https://doi.org/10.1016/j.dib.2018.11.015>

Dunphy, K., Sadhu, A., & Wang, J. (2022). "Multiclass Damage Detection in Concrete Structures Using a Transfer Learning-based Generative Adversarial Networks," *Structural*

Control and Health Monitoring, 29(11). <https://doi.org/10.1002/stc.3079>

Feroz, S., & Abu Dabous, S. (2021). "UAV-based Remote Sensing Applications for Bridge Condition Assessment," *Remote Sensing*, 13(9), 1809. <https://doi.org/10.3390/rs13091809>

FHWA. (1995). "Recording and Coding Guide - Federal Highway Administration," Federal Highway Administration, 124 pp. <https://www.fhwa.dot.gov/bridge/mtguide.pdf>

FHWA. (2022). "Specifications for the National Bridge Inventory," Federal Highway Administration, 354 pp. https://www.fhwa.dot.gov/bridge/snbi/snbi_march_2022_publication.pdf

Gucunski, N., & Nazarian, S. (2010). "Material Characterization and Condition Assessment of Reinforced Concrete Bridge Decks by Complementary NDE Technologies," In *Structures Congress 2010* (pp. 429-439). [https://doi.org/10.1061/41130\(369\)40](https://doi.org/10.1061/41130(369)40)

Gucunski, N., Imani, A., Romero, F., Nazarian, S., Yuan, D., Wiggenhauser, H., Shokouhi, P., Taffe, A. & Kutrubes, D. (2012). "Nondestructive Testing to Identify Concrete Bridge Deck Deterioration," *The National Academies Press*: Washington, DC, USA, 2012. <https://doi.org/10.17226/22771>

Harris, C., & Stephens, M. (1988, August). "A Combined Corner and Edge Detector,". In *Alvey Vision Conference*, 15(50), pp. 10–5244. <https://doi.org/10.5244/c.2.23>

Hiasa, S., Catbas, F. N., Matsumoto, M., & Mitani, K. (2016). "Monitoring Concrete Bridge Decks Using Infrared Thermography with High Speed Vehicles," *Structural Monitoring and Maintenance*, 3(3), 277–296. <https://doi.org/10.12989/smm.2016.3.3.277>

Ichi, E., & Dorafshan, S. (2022). "Effectiveness of Infrared Thermography for Delamination Detection in Reinforced Concrete Bridge Decks," *Automation in Construction*, 142, 104523. <https://doi.org/10.1016/j.autcon.2022.104523>

Ichi, E., Jafari, F., & Dorafshan, S. (2022). "SDNET2021: Annotated NDE Dataset for Subsurface Structural Defects Detection in Concrete Bridge Decks," *Infrastructures*, 7(9), 107. <https://doi.org/10.3390/infrastructures7090107>

Infobridge. (2024). Federal Highway Administration, Retrieved on Jan. 15, 2024: <https://infobridge.fhwa.dot.gov>.

Jahanshahi, M. R., Masri, S. F., Padgett, C. W., & Sukhatme, G. S. (2013). "An Innovative Methodology for Detection and Quantification of Cracks through Incorporation of Depth Perception," *Machine vision and applications*, 24, 227-241. <https://doi.org/10.1007/s00138-011-0394-0>

Jha, D., Smedsrud, P. H., Riegler, M. A., Johansen, D., De Lange, T., Halvorsen, P., & Johansen, H. D. (2019, December). "Resunet++: An Advanced Architecture for Medical Image

Segmentation,” In *2019 IEEE international symposium on multimedia (ISM)*, pp. 225-2255. IEEE. <https://doi.org/10.1109/ISM46123.2019.00049>

Jiang, S., Cheng, Y., & Zhang, J. (2023). “Vision-guided Unmanned Aerial System for Rapid Multiple-type Damage Detection and Localization,” *Structural Health Monitoring*, 22(1), 319-337. <https://doi.org/10.1177/14759217221084878>

Jiang, S., Zhang, J., Wang, W., & Wang, Y. (2023). “Automatic Inspection of Bridge Bolts Using Unmanned Aerial Vision and Adaptive Scale Unification-based Deep Learning,” *Remote Sensing*, 15(2), 328. <https://doi.org/10.3390/rs15020328>

Kao, S. P., Chang, Y. C., & Wang, F. L. (2023). “Combining the YOLOv4 Deep Learning Model with UAV Imagery Processing Technology in the Extraction and Quantization of Cracks in Bridges,” *Sensors*, 23(5), 2572. <https://doi.org/10.3390/s23052572>

Khan, F., & Bartoli, I. (2015). “Detection of Delamination in Concrete Slabs Combining Infrared Thermography and Impact Echo Techniques: A Comparative Experimental Study,” *Structural Health Monitoring and Inspection of Advanced Materials, Aerospace, and Civil Infrastructure 2015*, 94370I, pp. 101–111. <https://doi.org/10.1117/12.2084096>

Khudhair, M., & Gucunski, N. (2023). “Multi-NDE Technology Approach to Improve Interpretation of Corrosion in Concrete Bridge Decks Based on Electrical Resistivity Measurements,” *Sensors*, 23(19), 8052. <https://doi.org/10.3390/s23198052>

Kilic, G., & Caner, A. (2021). “Augmented Reality for Bridge Condition Assessment Using Advanced Non-destructive Techniques,” *Structure and Infrastructure Engineering*, 17(7), 977–989. <https://doi.org/10.1080/15732479.2020.1782947>

Koch, C., Georgieva, K., Kasireddy, V., Akinci, B., & Fieguth, P. (2015). “A Review on Computer Vision Based Defect Detection and Condition Assessment of Concrete and Asphalt Civil Infrastructure,” *Advanced engineering informatics*, 29(2), 196-210. <https://doi.org/10.1016/j.aei.2015.01.008>

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). “ImageNet Classification with Deep Convolutional Neural Networks,” *Advances in neural information processing systems*, 25. https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

La, H. M., Lim, R. S., Basily, B., Gucunski, N., Yi, J., Maher, A., Romero, F. A., & Parvardeh, H. (2013, August). “Autonomous Robotic System for High-efficiency Non-destructive Bridge Deck Inspection and Evaluation,” In *2013 IEEE International Conference on Automation Science and Engineering (CASE)* (pp. 1053–1058). IEEE

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). “Backpropagation Applied to Handwritten Zip Code Recognition,” *Neural computation*, 1(4), 541-551. <https://doi.org/10.1162/neco.1989.1.4.541>

- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). "Gradient-based Learning Applied to Document Recognition," *Proceedings of the IEEE*, 86(11), 2278-2324.
<https://doi.org/10.1109/5.726791>
- Lee, S., Kalos, N., & Shin, D. H. (2014). "Non-destructive Testing Methods in the US for Bridge Inspection and Maintenance," *KSCE Journal of Civil Engineering*, 18, 1322-1331.
<https://doi.org/10.1007/s12205-014-0633-9>
- Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011). "BRISK: Binary Robust Invariant Scalable Keypoints," In *2011 International Conference on Computer Vision*, pp. 2548–2555, IEEE.
<https://doi.org/10.1109/ICCV.2011.6126542>
- Li, X., Sun, X., Meng, Y., Liang, J., Wu, F., & Li, J. (2019). "Dice Loss for Data-imbalanced NLP Tasks," *arXiv preprint arXiv:1911.02855*. <https://doi.org/10.48550/arXiv.1911.02855>
- Long, J., Shelhamer, E., & Darrell, T. (2015). "Fully Convolutional Networks for Semantic Segmentation," In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431-3440.
https://openaccess.thecvf.com/content_cvpr_2015/html/Long_Fully_Convolutional_Networks_2015_CVPR_paper.html
- Lowe, D. G. (2004). "Distinctive Image Features from Scale-invariant Keypoints," *International Journal of Computer Vision*, 60, 91–110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- Mac, V. H., Huh, J., Doan, N. S., Lee, B. Y., & Haldar, A. (2021). "A Comprehensive Study on Identification of Both Deck and Soffit Defects in Concrete Bridge Decks through Thermographic Investigation of Shaded Side under Natural Conditions," *Construction and Building Materials*, 303, 124452. <https://doi.org/10.1016/j.conbuildmat.2021.124452>
- Mac, V. H., Tran, Q. H., Huh, J., Doan, N. S., Kang, C., & Han, D. (2019). "Detection of Delamination with Various Width-to-depth Ratios in Concrete Bridge Deck Using Passive IRT: Limits and applicability," *Materials*, 12(23), 3996. <https://doi.org/10.3390/ma12233996>
- Maeda, H., Sekimoto, Y., Seto, T., Kashiwayama, T., & Omata, H. (2018). "Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images," *Computer-Aided Civil and Infrastructure Engineering*, 33(12), 1127-1141.
<https://doi.org/10.1111/mice.12387>
- Ohio DOT. (2014). "Manual of Bridge Inspection," The Ohio Department of Transportation. Columbus, Ohio.
- Matsumoto, M., Mitani, K., & Catabas, F. N. (2014). "Non-destructive Bridge Deck Assessment Using Image Processing and Infrared Thermography," In *Transportation Research Board 93rd Annual Meeting*. https://www.pwri.go.jp/eng/ujnr/tc/g/pdf/29/29-3-5_Matsumoto.pdf

- Mirzazade, A., Popescu, C., Gonzalez-Libreros, J., Blanksvärd, T., Täljsten, B., & Sas, G. (2023). "Semi-autonomous Inspection for Concrete Structures Using Digital Models and a Hybrid Approach Based on Deep Learning and Photogrammetry," *Journal of Civil Structural Health Monitoring*, 13(8), 1633-1652. <https://doi.org/10.1007/s13349-023-00680-x>
- NBIS (2022). "Code of Federal Regulations. 23 Highways Part 650, Subpart C – National Bridge Inspection Standards," National Bridge Inspection Standards. <https://www.ecfr.gov/current/title-23/chapter-I/subchapter-G/part-650/subpart-C>
- Oh, J. K., Jang, G., Oh, S., Lee, J. H., Yi, B. J., Moon, Y. S., Lee, J.S., & Choi, Y. (2009). "Bridge Inspection Robot System with Machine Vision," *Automation in Construction*, 18(7), 929-941. <https://doi.org/10.1016/j.autcon.2009.04.003>
- Oh, T., Kee, S. H., Arndt, R. W., Popovics, J. S., & Zhu, J. (2012). "Comparison of NDT Methods for Assessment of a Concrete Bridge Deck," *Journal of Engineering Mechanics*, 139(3), 305–314. [https://doi.org/10.1061/\(ASCE\)EM.1943-7889.0000441](https://doi.org/10.1061/(ASCE)EM.1943-7889.0000441)
- Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., Glocker, B., & Rueckert, D. (2018). "Attention U-Net: Learning where to Look for the Pancreas," *arXiv preprint arXiv:1804.03999*. <https://doi.org/10.48550/arXiv.1804.03999>
- Omar, A., & Moselhi, O. (2022). "Condition Monitoring of Reinforced Concrete Bridge Decks: Current Practices and Future Perspectives," *Curr. Trends Civ. Struct. Eng*, 8, 1-4. <http://dx.doi.org/10.33552/CTCSE.2022.08.000695>
- Omar, T., Nehdi, M. L., & Zayed, T. (2018). "Infrared Thermography Model for Automated Detection of Delamination in RC Bridge Decks," *Construction and Building Materials*, 168, 313–327. <https://doi.org/10.1016/j.conbuildmat.2018.02.126>
- Pedram, M., Taylor, S., Hamill, G., Robinson, D., O'Brien, E. J., & Uddin, N. (2022). "Experimental Evaluation of Heat Transition Mechanism in Concrete with Subsurface Defects Using Infrared Thermography," *Construction and Building Materials*, 360, 129531. <https://doi.org/10.1016/j.conbuildmat.2022.129531>
- Perry, B. J., Guo, Y., Atadero, R., & Van De Lindt, J. W. (2020). "Streamlined Bridge Inspection System Utilizing Unmanned Aerial Vehicles (UAVs) and Machine Learning," *Measurement*, 164, 108048. <https://doi.org/10.1016/j.measurement.2020.108048>
- Pozzer, S., Rezazadeh Azar, E., Dalla Rosa, F., & Chamberlain Pravia, Z. M. (2021). "Semantic Segmentation of Defects in Infrared Thermographic Images of Highly Damaged Concrete Structures," *Journal of Performance of Constructed Facilities*, 35(1), 04020131. [https://doi.org/10.1061/\(ASCE\)CF.1943-5509.0001541](https://doi.org/10.1061/(ASCE)CF.1943-5509.0001541)
- Polycam. (n.d.). "Polycam - Lidar & 3D scanner for iPhone & Android," <https://poly.cam/>

- Prasanna, P., Dana, K., Gucunski, N., & Basily, B. (2012, April). "Computer-vision Based Crack Detection and Analysis," In *Sensors and smart structures technologies for civil, mechanical, and aerospace systems 2012*, 8345, pp. 1143-1148. SPIE. <https://doi.org/10.1117/12.915384>
- Raja, B. N. K., Miramini, S., Duffield, C., Sofi, M., & Zhang, L. (2022). "Infrared Thermography Detection of Delamination in Bottom of Concrete Bridge Decks," *Structural Control and Health Monitoring*, 29(3), e2886. <https://doi.org/10.1002/stc.2886>
- Robison, T. W., Barnes, C. L., Tinkey, Y., & Tanner, J. E. (2020). "Evaluating Concrete Damage in Bridge Decks with and without Overlays Using Nondestructive Testing Procedures," *Journal of Testing and Evaluation*, 48(1), 352–367. <https://doi.org/10.1520/JTE20180555>
- Rocha, J. H. A., Póvoas, Y. V., & Santos, C. F. (2019). Detection of Delaminations in Sunlight-unexposed Concrete Elements of Bridges Using Infrared Thermography," *Journal of Nondestructive Evaluation*, 38, 1–12. <https://doi.org/10.1007/s10921-018-0546-5>
- Rosten, E., Reitmayr, G., & Drummond, T. (2005, December). "Real-time Video Annotations for Augmented Reality," In *International Symposium on Visual Computing*, pp. 294–302. Springer. https://doi.org/10.1007/11595755_36
- Ronneberger, O., Fischer, P., & Brox, T. (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation," In *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pp. 234-241. Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28
- Rubio, J. J., Kashiwa, T., Laiteerapong, T., Deng, W., Nagai, K., Escalera, S., Nakayama, K., Matsuo, Y., & Prendinger, H. (2019). "Multi-class Structural Damage Segmentation Using Fully Convolutional Networks," *Computers in Industry*, 112, 103121. <https://doi.org/10.1016/j.compind.2019.08.002>
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). "ORB: An Efficient Alternative to SIFT or SURF," In *2011 International Conference on Computer Vision*, pp. 2564–2571, IEEE. <https://doi.org/10.1109/ICCV.2011.6126544>
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). "MobileNetV2: Inverted Residuals and Linear Bottlenecks," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520. <https://doi.org/10.48550/arXiv.1801.04381>
- Shi, Y., Cui, L., Qi, Z., Meng, F., & Chen, Z. (2016). "Automatic Road Crack Detection Using Random Structured Forests," *IEEE Transactions on Intelligent Transportation Systems*, 17(12), 3434-3445. <https://doi.org/10.1109/TITS.2016.2552248>
- Simonyan, K., & Zisserman, A. (2014). "Very Deep Convolutional Networks for Large-scale Image Recognition," *arXiv*. <https://doi.org/10.48550/arXiv.1409.1556>

Ta, Q. T., Huh, J., Yim, H. J., & Lee, G. (2025). "Automatic Detection of Subsurface Defects in Concrete Structures Using State-of-the-art Deep Learning-based Object Detectors on the Infrared Dataset," *Engineering Structures*, 329, 119829. <https://doi.org/10.1016/j.engstruct.2025.119829>

Tarussov, A., Vandry, M. & De La Haza, A. (2013). "Condition Assessment of Concrete Structures Using a New Analysis Method: Ground-penetrating Radar Computer-assisted Visual Interpretation," *Construction and Building Materials*, 38, pp.1246-1254. <https://doi.org/10.1016/j.conbuildmat.2012.05.026>

Tazarv, M., Won, K., Jang, Y., Hart, K. & Greenway, E. (2022). "Post-earthquake Serviceability Assessment of Standard RC Bridge Columns Using Computer Vision and Seismic Analyses," *Engineering Structures*, 272, p.115002. <https://doi.org/10.1016/j.engstruct.2022.115002>

Tomar, N. K., Shergill, A., Rieders, B., Bagci, U., & Jha, D. (2022). "TransResU-Net: Transformer Based ResU-Net for Real-time Colonoscopy Polyp Segmentation," *arXiv preprint arXiv:2206.08985*. <https://doi.org/10.48550/arXiv.2206.08985>

Tran, Q. H., Huh, J., Kang, C., & Han, D. (2018). "Effects of Rebars on the Detectability of Subsurface Defects in Concrete Bridges Using Square Pulse Thermography," *NDT & E International*, 100, 92–100. <https://doi.org/10.1016/j.ndteint.2018.09.001>

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). "Attention is all you need," *Advances in Neural Information Processing systems*, 30. <https://doi.org/10.48550/arXiv.1706.03762>

Vyas, V., Patil, V. J., Singh, A. P., & Srivastava, A. (2019). "Application of Infrared Thermography for Debonding Detection in Asphalt Pavements," *Journal of Civil Structural Health Monitoring*, 9, 325–337. <https://doi.org/10.1007/s13349-019-00337-8>

Washer, G., Fenwick, R., & Bolleni, N. (2010). "Effects of Solar Loading on Infrared Imaging of Subsurface Features in Concrete," *Journal of Bridge Engineering*, 15(4), 384–390. [https://doi.org/10.1061/\(ASCE\)BE.1943-5592.0000117](https://doi.org/10.1061/(ASCE)BE.1943-5592.0000117)

Wiggenhauser, H. (2008). "Advanced NDT Methods for the Assessment of Concrete Structures," In *Concrete Repair, Rehabilitation and Retrofitting II* (pp. 37-48). CRC Press. <http://dx.doi.org/10.1201/9781439828403.ch3>

Wisconsin DOT. (2024). "Structure Inspection Field Manual," Wisconsin Department of Transportation. *Madison, WI*.

Won, K., & Sim, C. (2020). "Automated Transverse Crack Mapping System with Optical Sensors and Big Data Analytics," *Sensors*, 20(7), 1838. <https://doi.org/10.3390/s20071838>

Zeiler, M. D., & Fergus, R. (2014). "Visualizing and Understanding Convolutional Networks,"

In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part I* 13, pp. 818–833. Springer International Publishing.
https://doi.org/10.1007/978-3-319-10590-1_53

Zhang, C., Chang, C. C., & Jamshidi, M. (2020). “Concrete Bridge Surface Damage Detection Using a Single-stage Detector,” *Computer-Aided Civil and Infrastructure Engineering*, 35(4), 389–409. <https://doi.org/10.1111/mice.12500>

Zhang, Z., Liu, Q., & Wang, Y. (2018). “Road Extraction by Deep Residual U-Net,” *IEEE Geoscience and Remote Sensing Letters*, 15(5), 749–753.
<https://doi.org/10.1109/LGRS.2018.2802944>

Zhang, L., Yang, F., Zhang, Y. D., & Zhu, Y. J. (2016, September). "Road Crack Detection Using Deep Convolutional Neural Network," In *2016 IEEE international conference on image processing (ICIP)* (pp. 3708–3712). IEEE. <https://doi.org/10.1109/ICIP.2016.7533052>

Zheng, D., Tan, S., Li, X., & Cai, H. (2021). “Research on the Infrared Thermographic Detection of Concrete under Solar Heating,” *Advances in Civil Engineering, 2021*, Article ID 6692729, 1–10. <https://doi.org/10.1155/2021/6692729>

Zhu, Z., German, S., & Brilakis, I. (2010). “Detection of Large-scale Concrete Columns for Automated Bridge Inspection,”. *Automation in construction*, 19(8), 1047–1055.
<https://doi.org/10.1016/j.autcon.2010.07.016>

Zou, Q., Zhang, Z., Li, Q., Qi, X., Wang, Q., & Wang, S. (2018). “Deepcrack: Learning Hierarchical Convolutional Features for Crack Detection,” *IEEE transactions on image processing*, 28(3), 1498–1512.
<https://doi.org/10.1109/TIP.2018.2878966>