GEORGIA DOT RESEARCH PROJECT 22-01

Final Report

UAS-ASSISTED INSPECTION OF BRIDGES FOR CORROSION EFFECTS

Office of Performance-based Management and Research
600 West Peachtree Street NW | Atlanta, GA 30308

December 2023

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. FHWA-GA-23-2201	2. Government Accession No. N/A	3. Recipient's Catalog No. N/A
4. Title and Subtitle UAS-Assisted Inspection of Bridges for Corrosion Effects		5. Report Date December 2023
		6. Performing Organization Code N/A
7. Author(s) Iris Tien, Ph.D. (PI) (https://orcid.org/0000-0002-1410-632X) Hana Herndon (Graduate Research Assistant)		8. Performing Organization Report No. 22-01
9. Performing Organization Name and Address Georgia Institute of Technology School of Civil and Environmental Engineering 790 Atlantic Drive Atlanta, GA 30332		10. Work Unit No. N/A
		11. Contract or Grant No. PI#0017924
12. Sponsoring Agency Name and Address Georgia Department of Transportation (SPR) Office of Performance-based Management and Research 600 West Peachtree St. NW Atlanta, GA 30308		13. Type of Report and Period Covered Final Report (August 2022 – December 2023)
		14. Sponsoring Agency Code: N/A

15. Supplementary Notes

Prepared in cooperation with the U.S. Department of Transportation, Federal Highway Administration.

16. Abstract

New technologies, such as unmanned aerial vehicles (UAVs) and machine learning, have been used to conduct imagery-based bridge inspections and evaluate damage on bridges. However, corrosion detection is still an open problem, and corrosion detection algorithms have only proven adequate in certain environments and conditions. The main goal of this project is to explore the use of UAVs as a proof of concept to detect and characterize corrosion on bridges in Georgia. The first objective of this research is to investigate the use of UAVs for bridge inspections. The second objective is to develop and evaluate automatic corrosion detection and evaluation algorithms. As part of this project, imagery data from two bridges, selected in collaboration with Georgia Department of Transportation (GDOT) personnel, was collected using the Skydio 2+ drone. After the images were cleaned and labeled, they were used in varying computer vision and machine learning algorithms for corrosion detection. First, texture thresholding and color thresholding methods were implemented. Then, K-means was investigated as an unsupervised machine learning algorithm using texture and color features. Finally, deep learning methods were investigated with automated feature extraction. It was found that the Skydio 2+ drone can be used to reach places that are hard for inspectors to reach. However, vegetation surrounding the bridge limits where the drone can fly and how much of the bridge can be assessed. Therefore, pre-flight planning is recommended. Additionally, it was found that no corrosion segmentation technique as of yet works perfectly. K-means segmentation, coupled with background removal, has the highest recall, but the mIoU is low. Texture thresholding performed with the second highest recall and an mIoU close to 90 percent. The deep learning techniques, Spot Rust and FCN, were found to have the lowest performance. This research shows the promise of these methodologies to assist in bridge inspections along with the need to further develop them. It provides results that serve as a good basis to further investigate and develop machine learning methods to use UAVs in support of bridge inspections in Georgia.

17. Key Words Corrosion, computer vision, machine learning, structural health monitoring, bridge inspection, image segmentation	18. Distribution Statement No restrictions. This document is available through the National Technical Information Service, Springfield, VA 22161.

19. Security Classification (of this report) Unclassified	20. Security Classification (of this page) Unclassified	21. No. of Pages 62	22. Price Free

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

GDOT Research Project 22-01

Final Report

UAS-ASSISTED INSPECTION OF BRIDGES FOR CORROSION EFFECTS

By

Iris Tien, Ph.D.
Associate Professor

Hana Herndon
Graduate Research Assistant

Georgia Tech Research Corporation

Contract with
Georgia Department of Transportation

In cooperation with
U.S. Department of Transportation
Federal Highway Administration

December 2023

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	$candela/m^2$	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	$candela/m^2$	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	0.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

* SI is the symbol for the International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

CNN	Convolutional Neural Network
CVAT	Computer Vision Annotation Tool
FCN	Fully Convolutional Network
GDOT	Georgia Department of Transportation
GLCM	Gray-level Co-variance Matrix
GPS	Ground Positioning System
GPU	Graphics Processing Unit
HSV	Hue-Saturation-Value
IoU	Intersection Over Union
L*ab	Lightness-a-b
mIoU	Mean Intersection Over Union
RGB	Red-Green-Blue
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle

EXECUTIVE SUMMARY

New technologies, such as unmanned aerial vehicles (UAVs) and machine learning, have been used to conduct imagery-based bridge inspections and efficiently evaluate damage on bridges. However, corrosion detection is still an open problem, and corrosion detection algorithms have only proven adequate in certain environments and conditions. The main goal of this project is to explore the use of UAVs as a proof of concept to detect and characterize corrosion on bridges in Georgia. The first objective of this research is to investigate the use of UAVs for bridge inspections. The second objective is to develop and evaluate automatic corrosion detection and evaluation algorithms.

As part of this project, imagery data from two bridges, selected in collaboration with Georgia Department of Transportation (GDOT) personnel, were collected using the Skydio 2+ drone. After the images were cleaned and labeled, they were used in varying computer vision and machine learning algorithms for corrosion detection. First, texture thresholding and color thresholding methods were implemented. Then, K-means was investigated as an unsupervised machine learning algorithm using texture and color features. Finally, deep learning methods were investigated with automated feature extraction.

It was found that the Skydio 2+ drone can be used to reach places that are hard for inspectors to reach, but surrounding vegetation can limit where the drone can fly and how much of the bridge can be assessed. Therefore, pre-flight planning is recommended. Additionally, it was found that no corrosion segmentation technique as of yet works perfectly. K-means segmentation, coupled with background removal, has the highest recall, but the mean intersection over union (mIoU) is low. Texture thresholding performed with the second highest recall and an mIoU close to

90 percent. The deep learning techniques, SpotRust and Fully Convolutional Network (FCN), were found to have the lowest performance. This research shows promise in using these methodologies to assist in bridge inspections, as well as the need to further develop them. It provides results that serve as a good basis to further investigate and develop machine learning methods to use UAVs in support of bridge inspections in Georgia.

CHAPTER 1. INTRODUCTION

MOTIVATION

Properly functioning transportation networks are essential not only for the socioeconomic welfare of people and communities, but they are also essential for recovery after a disaster. One of the most vital components of transportation networks are bridges because they allow access to places that otherwise cannot be reached. Structural and functional losses to bridges can disrupt traffic across an entire network in ways that losses of other network elements will not. Therefore, it is crucial to be able to understand the state of each bridge asset to help engineers decide and prioritize which structures to repair and reconstruct before structural losses occur.

To understand the states of their assets, the Georgia Department of Transportation (GDOT) conducts routine bridge inspections every 24 months and damage inspections when necessary, which results in 8,414 bridge inspections annually.[1] However, there are several opportunities to improve upon and facilitate current bridge inspection methods. First, bridge inspections can be inefficient because they require hours of manual labor and road closures to provide access for inspectors, thereby disrupting the transportation network. Second, bridge inspections can be unsafe because inspectors are typically suspended by cables under what can be precarious conditions to view all components of a bridge, putting inspectors at risk. Finally, the information obtained from inspections may be incomplete if inspectors cannot reach all components of the bridge.

New technologies have the opportunity to address some of these limitations. For example, unmanned aerial systems (UASs) and unmanned aerial vehicles (UAVs) have been used to conduct imagery-based inspections. Because inspectors no longer need to be suspended to

inspect the bridges, UAVs can be used to improve inspector safety. Moreover, the bridge does not need to be closed for hours, improving the efficiency of the inspection process. However, without rigorous methodologies for processing the UAV-collected imagery data, using UAVs alone does not address the potential for inaccurate information and still requires hours of manual inspection time in working with the UAV data.

Recently, studies have attempted to address the problem of inaccurate or vague information provided by UAVs by using machine learning algorithms to characterize cracks in concrete bridges and corrosion in steel bridges from drone images with high accuracy.[2,3] Compared to crack detection, corrosion detection and characterization is still an open problem. In previous literature, the corrosion detection algorithms have only proven adequate in certain environments and conditions. Corrosion is particularly critical for bridges, as it is one of the most common causes of bridge failure, along with scour and construction mistakes.[4] Therefore, it is important that corrosion is comprehensively and accurately detected and characterized during bridge inspections and damage inspections.

RESEARCH OBJECTIVE

The main goal of this project is to explore the use of UAVs as a proof of concept to detect and characterize corrosion on bridges. This project has undertaken two main objectives to accomplish this goal. The first objective of this research is to investigate the use of UAVs for bridge inspections. The investigation includes whether UAVs can be used to inspect all components of the bridge, what limitations exist when using UAVs for bridge inspections, and how UAVs can assist human inspectors when making inspection decisions. The second objective of this research is to develop and evaluate automatic corrosion detection and evaluation

algorithms. Through this evaluation, we determine whether we can automatically detect corrosion from UAV images collected in Georgia and in what environmental conditions. In addition, this research will determine which corrosion evaluation algorithms have the best performance and are the most computationally efficient for use in the bridge inspection process.

CHAPTER 2. UAV IMAGERY DATA COLLECTION

CHALLENGES WITH UAV IMAGERY

The advent of affordable UAVs that can carry digital cameras close to structures presents an opportunity to conduct quick, safe, and effective bridge inspections from a distance while helping engineers understand the state of the structure and make repair decisions that protect against failure from corrosion. UAVs are increasingly used for civil inspections because they are noncontact, time-efficient, and cost-efficient, and can help inspectors see angles or objects that are difficult to view from the ground. Because bridge inspectors would no longer need to be suspended to inspect the bridges, UAVs can improve inspector safety during the inspection. Moreover, the bridge does not need to be closed for hours, which improves the efficiency of the inspection process.

Although UAVs can improve the safety and efficiency of bridge inspections, the images collected from UAVs can be difficult to analyze for damage because they capture many misleading objects. Examples of such misleading objects include graffiti, insect hives, and background vegetation, among others, all of which have significant lighting irregularities. Additionally, in UAV images, a larger portion of the image will not represent corrosion compared to those images collected by humans. Whereas a human inspector will only take an image when corrosion is present and observed, a drone will take an image regardless of the bridge condition. This difference in proportion of corrosion in a given image makes it more difficult to notice and characterize. Figure 1 shows the differences between photographs captured by a human inspector and photographs captured from a UAV. For these reasons, corrosion can

be more difficult to detect in UAV-collected images than in human-collected images and field inspections.

(a) (b)

Figure 1. Photographs. (a) Human-collected[5] versus (b) drone-collected images of corrosion on bridges.

DRONE EQUIPMENT SELECTION: SKYDIO 2+ ENTERPRISE

The drone selected to collect data and test UAV-aided bridge inspections is the Skydio 2+ Enterprise drone. This drone was selected in collaboration with GDOT personnel. Some unique features of this model include vision-based obstacle avoidance, 180-degree gimbal movement, automatic photo capture, and 3× zoom. These features provide many advantages, as well as certain limitations, during bridge inspections.

A vision-based obstacle avoidance system is a primary advantage of the Skydio 2+ model. Most other models use a Global Positioning System (GPS)–based obstacle avoidance system and

therefore require a strong GPS signal to fly. The Skydio 2+ model, on the other hand, does not require a strong GPS signal to fly and can be used to collect data in more locations. This feature is especially important for bridge inspections because many bridges are located in areas with low GPS signals and a GPS-based model could not be used to collect inspection data for those bridges. Additionally, the Skydio 2+ drone can avoid obstacles with greater success than GPS-based models because it combines photometric information with scene understanding in real-time using deep learning algorithms.[6] However, like other models, the Skydio 2+ model is not capable of avoiding thin wires or branches. This is a major limitation to overcome in Georgia because most bridges are surrounded by trees that block the available flying space of the drone and how much of the bridge can be captured. This limitation is discussed further in the next two sections, which discuss the data collection at Bear Creek Bridge, Douglasville, Georgia, and Salacoa Creek Bridge, Calhoun, Georgia. One disadvantage to the Skydio 2+ model's use of vision-based obstacle avoidance is that it will only take off when there is adequate lighting; therefore, it will not fly during dusk or dawn. This is an important limitation to note for bridge inspections, which would be better conducted when there is lower traffic, particularly if any road closures are required. However, this limitation is an inconvenience rather than an obstacle and can be overcome for use in the bridge inspection process through scheduling. Ultimately, any required road closures during a UAV-aided bridge inspection would take less time than a road closure for a traditional bridge inspection, so the time efficiency of the inspection would still improve.

Another advantage of the Skydio 2+ model is the 180-degree gimbal movement. This feature allows the UAV to capture straight images of the underside and overside of the horizontal bridge deck. Other UAV models that do not have this feature can only collect images of the bridge deck

at an angle, which impacts the accuracy of automatic damage detection and evaluation on the photographs.

The Skydio 2+ drone can easily capture high-quality photographs due to the automatic image capture and 3× zoom. The automatic image capture allows inspectors to collect images without pausing the flight and makes the inspection process more efficient. The images captured automatically are high quality (i.e., resolution) and do not have blurring up to a certain speed even if the drone is in motion. The frequency of the image capture can be configured in the drone settings, so inspectors can choose a setting that gives them comprehensive data without being too repetitive. The 3× zoom capability of the Skydio 2+ model also improves the image capture process because it makes bridge components and damage more visible. The zoom can significantly reduce how much of the captured photograph is background chaos, particularly foliage in the background, which improves the accuracy of corrosion detection and evaluation. However, it is important to note that photographs taken with the zoom will still have variable lighting, and therefore still need to be preprocessed before damage detection. This process is discussed further in the sections on Cleaning and Color Thresholding.

Another advantage of the Skydio 2+ model for bridge inspections is that it can take off and land on the Skydio 2+ case or in the inspector's hand. This feature is especially helpful for bridge inspections in the mountainous regions of Georgia where there may not be a flat clearing to take off and land near the bridge. Therefore, it increases the number of places where the Skydio drone can be used for bridge inspections.

A notable disadvantage of the Skydio 2+ drone is that its flight time on a single battery is 25 minutes. The Enterprise Package comes with four batteries, which allows for a total of 1 hour

and 40 minutes of flight time for bridge inspections. To mitigate this limitation, Skydio software has features that remember where the drone ended its last flight so it can start in the same position on the next battery. It is important to note that this flight time is consistent with other drone models on the market, and therefore would not be reduced with another model. However, this flight time may not be adequate to collect comprehensive data on larger bridges in Georgia.

BEAR CREEK BRIDGE, DOUGLASVILLE, GEORGIA

As part of this project, imagery data from two bridges were collected using the Skydio 2+ drone. The bridges were selected in collaboration with GDOT personnel. The first bridge that was inspected was GDOT #097-0013-0. This bridge is located in Douglasville, Georgia, along State Route (SR) 166 and crosses Bear Creek.

The bridge inspection with the Skydio 2+ drone took place on December 12, 2022. The weather was overcast, so the lighting variations in the images were minimal. There was right-of-way available to take off and land the drone from the ground (see figure 2). In total, the inspectors took 722 photographs using many features of the Skydio 2+, including automatic image capture and 180-degree gimbal rotation.

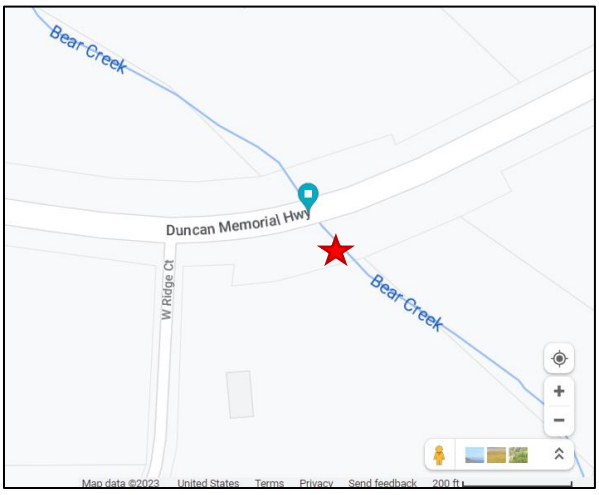

Figure 2. Map. Piloting location at Bear Creek.

There were many trees and thin branches at this site, which limited where the drone could fly, what sections of the bridge it could capture, and therefore the completeness of the data (see figure 3). These photographs are examples of the images taken during the test inspection. As shown in the figure, the north side of the bridge was almost completely blocked by trees. To avoid collisions, no images were attempted from this angle.

Although figure 3 shows visible corrosion (red circles) in images taken on the site, the lighting variations limit the view in portions of the bridge that are out of focus in the image. For the piles and beams in the foreground of the image, the lighting is generally adequate for the human eye to view corrosion. These photographs were taken using automatic image capture while the drone was in motion at 2–10 mph speeds.

Figure 3. Photographs. Bridge over Bear Creek, Douglasville.
Red circles = identified corrosion.

In these photographs, the majority of the image space is dedicated to the background, and the object of interest, i.e., the corrosion shown in the red circles, takes up a relatively small amount of image space. Further, because the photographs were taken in December, the leaves on the ground are a brown-red color that resembles corrosion more closely than foliage that would be present during spring or summer. This resemblance made automatic corrosion detection more difficult, especially for the image processing algorithms, as discussed in chapter 4.

Figure 4 shows the underside of the bridge deck with different degrees of gimbal rotation. The Skydio 2+ drone is capable of collecting photographs that give a complete understanding of the underside of the bridge deck due to the 180-degree gimbal rotation feature. In these photographs, the lighting improves as the gimbal rotation increases because the lighting from the sky is in less of the photograph; the Skydio camera adjusts accordingly, leading to better visibility of the corroded areas.

Figure 4. Photographs. Underside of the Bear Creek bridge deck at (a) 0° rotation (b) 50° rotation and (c) 80° rotation.

SALACOA CREEK BRIDGE, CALHOUN, GEORGIA

The second bridge inspected using the Skydio 2+ drone was GDOT #129-0048-0, a bridge located in Calhoun, Georgia, along SR 156 where it crosses Salacoa Creek.

The bridge inspection with the Skydio 2+ drone took place on February 20, 2023. The weather was partially cloudy, which resulted in significant lighting and reflectance variations in the collected images. There was no right-of-way available to take off, so the inspectors hiked under the bridge to allow takeoff from a location away from traffic (see figure 5). There was also no flat space, so the ability to take off and land from the case allowed this test inspection to take place. In total, the inspectors took 1171 photographs using many features of the Skydio 2+, including automatic image capture, 3× zoom, and 180-degree gimbal rotation.

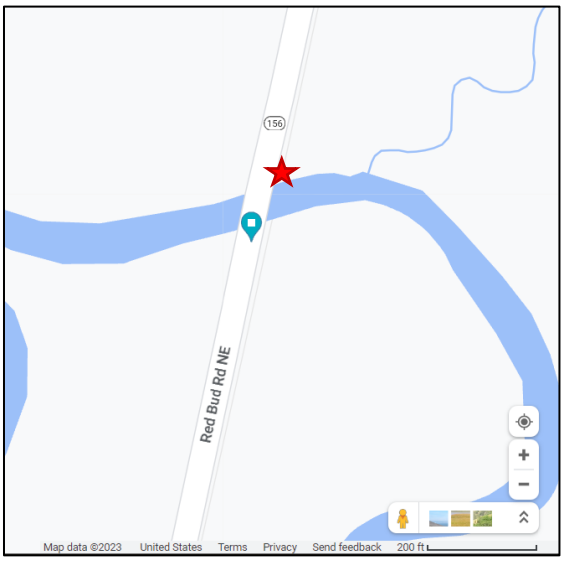

Figure 5. Map. Piloting location at Salacoa Creek.

Similar to Bear Creek, many thin branches at this site limited where the drone could fly (see figure 6). These photographs are examples of the images taken during the inspection. As shown, the portion of the bridge that crossed dry land rather than the creek was almost entirely unreachable due to the thin trees and branches present. Therefore, the inspectors were only able to take images of the northern half of the bridge.

Figure 6 also shows the variable lighting that was present due to the weather conditions. In the photograph on the left, the majority of the bridge is covered in shadows, hiding most of the corrosion and rendering this photograph unusable for automatic corrosion detection. The photograph on the right shows an example where the lighting was better and corrosion is visible. These photographs were taken using automatic image capture while the drone was in flight at speeds ranging 2–10 mph.

Figure 6. Photographs. Bridge over Salacoa Creek, Calhoun.

Again, as seen in figure 6, the majority of the image space is dedicated to the background, so the object of interest, i.e., corrosion, takes up the least amount of image space. Although there was not as much foliage on the ground, the coloring of the soil at Salacoa Creek resembles corrosion more closely than the soil at Bear Creek, which caused confusion for the automatic corrosion detection algorithms, which will be discussed further in chapter 4.

Figure 7 shows the increased visibility in images taken with 3× zoom compared to images taken with standard magnification. In the standard image, it is difficult for humans to determine if the corrosion present on the left pile is actual corrosion or soil stuck on the structure. However, in the zoomed photograph, it is easier to determine that it is indeed corrosion. It is important to note, as well, how the lighting improves in the zoomed photograph versus the standard photograph. Because the Skydio 2+ automatically adjusts lighting according to the composition of the photograph, when less sky and other bright objects are present, the shadows become more uniform and the corroded areas are easier to observe.

(a) (b)

Figure 7. Photographs. (a) Standard and (b) zoomed view of corroded pile at Salacoa Creek.

CHAPTER 3. DATA CLEANING AND LABELING

To prepare the imagery data collected from the drone flights for image segmentation, we conducted the following activities: contrast enhancement, image augmentation, background removal, and image labeling. These data cleaning and labeling activities were conducted to improve the performance of various corrosion segmentation algorithms. Contrast enhancement increases the range of values in grayscale images, which can improve texture detection and thresholding methods. Image augmentation increases the size and variability of the dataset, which improves the performance of deep learning algorithms. Background removal was conducted after we discovered that computer vision techniques yield many false positives in the background of the images because the foliage resembles corrosion in texture and color. Lastly, the images were labeled to prepare them for training a deep learning algorithm, which requires ground truth labels to learn the features of the dataset and make predictions. Each of these activities is described in more detail in the following sections.

CLEANING

To clean the data and prepare it for image segmentation, we enhanced the contrast of the images using histogram equalization, augmented the dataset using color jitter and horizontal flipping functions, and removed the background using a deep learning algorithm trained on data collected at Bear Creek and Salacoa Creek. These data cleaning methods improved corrosion detection to varying degrees, as described in chapter 4.

Contrast Enhancement

Two potential methods to enhance the contrast in an image are histogram equalization and contrast stretching. Histogram equalization spreads out the most frequent intensity values in an

image so the cumulative distribution function of intensity in the image is roughly linear. However, this method can oftentimes yield unnatural results. Contrast stretching, on the other hand, stretches the range of intensity values linearly across a wider range. This method yields more natural results, but the intensity values are not as dispersed as with histogram equalization (see figure 8).[7] When detecting corrosion from images, the intensity values of the pixels are more important than the visual appearance of the image. Therefore, histogram equalization was applied to images during texture thresholding. The results of corrosion detection with and without histogram equalization are shown and discussed in Texture Thresholding.

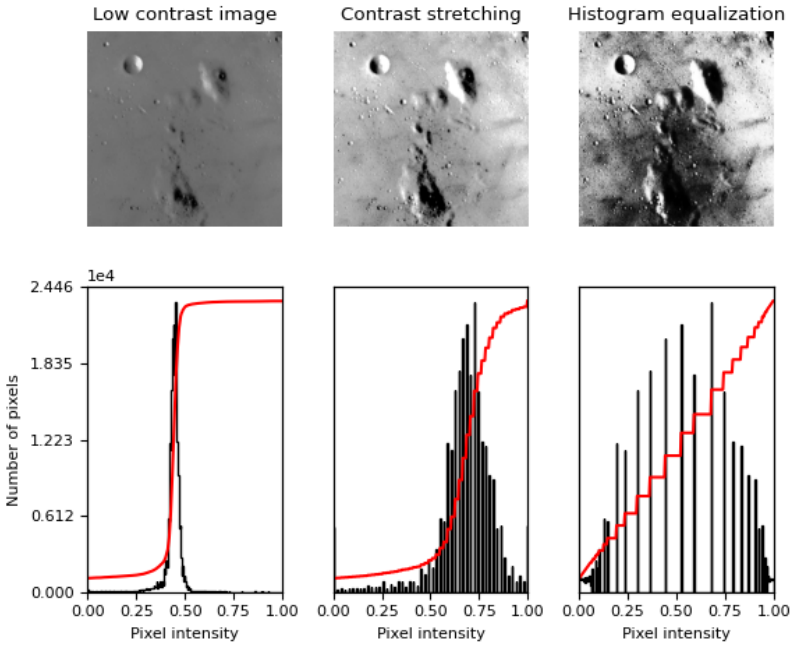

Figure 8. Photographs and charts. Histogram equalization (right) versus contrast stretching (center) on a low-contrast image (left).[7]

Image Augmentation

Image augmentation is a method used to improve the performance of deep learning algorithms, which need large amounts of training data to achieve good performance. Image augmentation

generates new transformed versions of images from the given dataset. This not only increases the number of training samples available but also diversifies the dataset. In this study, the augmentation methods utilized were random horizontal flip and color jitter, which were conducted repeatedly to increase the size of the dataset by seven times.

Background Removal

After conducting image processing on the images, it was determined that the background was too chaotic, and this chaos severely impacted the results of the algorithms. The background consisted mostly of foliage that was textured and red-brown colored, which resembled corrosion from a computer vision point of view. Therefore, when using the texture or color characteristics to segment corrosion, the algorithms also segmented the background. It was determined that the background needed to be removed to improve the results of the image processing algorithms. This process is discussed more in chapter 4 in the sections on Texture Thresholding, Color Thresholding, and K-Means Algorithm.

The background was removed using a deep learning algorithm. Although there are open-source background removal algorithms available, they are not trained on images of bridges and therefore do not perform well on the dataset. Therefore, the PyTorch implementation of the fully convolutional network (FCN) architecture was trained using the data collected at Bear Creek and Salacoa Creek. Once the algorithm is trained, it can remove the background from the images quickly and accurately. Figure 9 and figure 10 show the mean Intersection over Union (IoU) history and loss history as the algorithm is trained to remove the background from images of bridges. A perfect performance would be 1.00 IoU and 0.00 loss, and the algorithms were run until convergence.

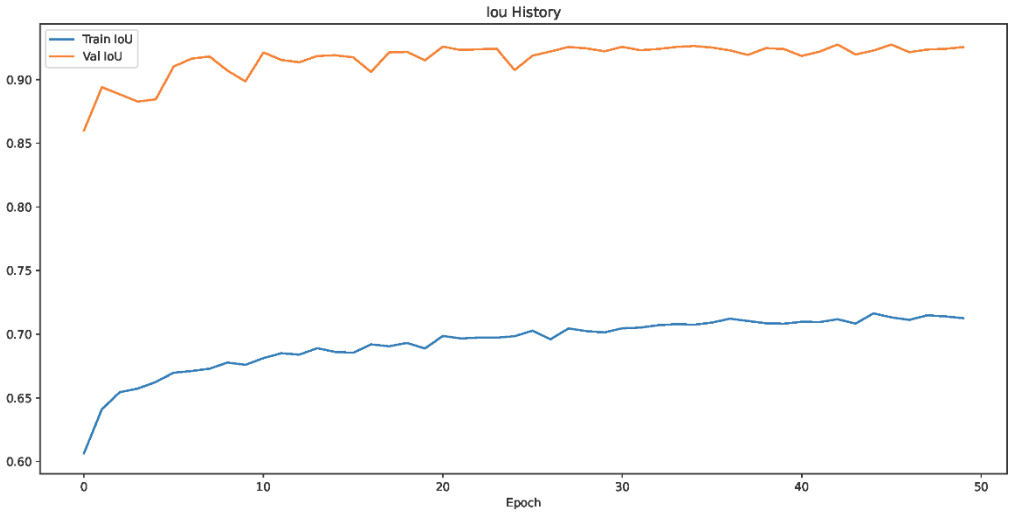

Figure 9. Graph. IoU history of background removal algorithm.

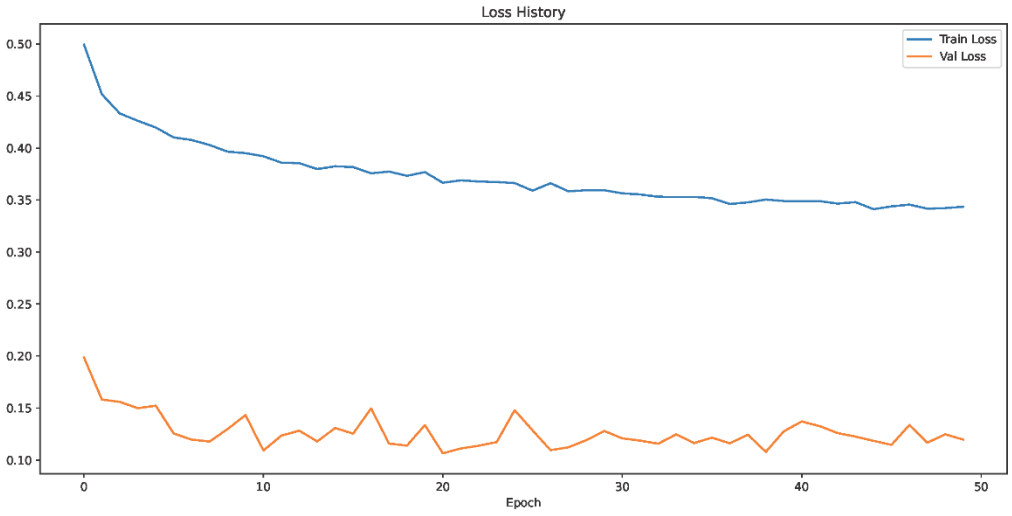

Figure 10. Graph. Loss history of background removal algorithm.

The final IoU of this model was 0.721 on the training dataset and 0.925 on the validation dataset. The final loss of this model was 0.336 on the training dataset and 0.120 on the validation dataset. These results suggest that although this model can remove the background from images of bridges well, it will not remove the background perfectly.

LABELING

The Computer Vision Annotation Tool (CVAT) was used to label the images. CVAT is a free, open-source, web-based image annotation tool that allows users to label data for all tasks of supervised machine learning.[8]

Five classes were labeled in the images:

- Background: any part of the photograph that is not part of the structure.
- Bridge: any part of the photograph that is part of the structure but is not corroded.
- Staining: any part of the bridge that has been stained due to corrosion leaking but does not affect the structure other than the appearance.
- Surface corrosion: corrosion that impacts the structure in appearance and capacity and is recognizable with its brown-red color and rough texture.
- Structural corrosion: corrosion that has reduced the section or material properties of the member.

Of the original 1893 images, 767 were labeled. Many images were discarded due to repetition or inadequate image quality. The quantity of labeled data is important to train the deep learning corrosion-detection algorithms but is not necessary for other computer vision methods. Through image augmentation, the final dataset was increased to 5369 images for use in the deep learning algorithm.

CHAPTER 4. COMPUTER VISION AND MACHINE LEARNING ALGORITHMS FOR CORROSION DETECTION

With the images cleaned and labeled, they were then used in varying computer vision and machine learning algorithms for corrosion detection. In general, digital image processing uses computer algorithms to extract information from digital images. It can include a wide range of algorithms, including machine learning and deep learning, but also encompasses non-artificial intelligence approaches such as color thresholding, color space optimization, image compression, entropy quantification, and texture thresholding. For the task of corrosion identification, researchers have used color and texture techniques to capitalize on the visual properties of corrosion, particularly texture and color, because corrosion produces a unique rough surface and color in most materials, including steel.[9] Texture can be evaluated using gray-level co-variance matrices (GLCMs), entropy quantification, or edge detection,[10] and color can be evaluated by thresholding the image in different color spaces. Both have been used in the evaluation of corrosion in images and were investigated in this project. First, texture thresholding and color thresholding methods were implemented. Then, K-means clustering was investigated as an unsupervised machine learning algorithm using texture and color features. Finally, deep learning methods were investigated with automated feature extraction.

TEXTURE THRESHOLDING

The texture was evaluated in the images collected at Bear Creek and Salacoa Creek using entropy quantification and edge detection. Entropy is the statistical measure of randomness, which measures the information or “surprise” of a pixel being a certain value. If an event is highly likely, it is not surprising and does not provide researchers with as much information as an unlikely or surprising event. A likely event has less entropy than an unlikely event. In the case

of texture quantification, the event in question is the value of a pixel. In textured regions of the image, the likelihood of a pixel being a specific value is lower than in smooth regions of the image. Therefore, high entropy regions represent texture or corrosion. Edge detection, or gradient operators, find the edges in an image using different types of convolution filters. These filters bring emphasis to regions where the pixel values change rapidly. Textured portions have more pixel value variation, and are therefore expected to have more edges than smooth ones.

Figure 11 and figure 12 show the entropy analysis of images collected at Bear Creek and Salacoa Creek, respectively. As seen in the photographs, the entropy was highest where there was foliage present and second highest where there was corrosion. The red and blue circles highlight the corrosion in the images.

Figure 11. Photographs. Entropy of Bear Creek images. Circles = corrosion.

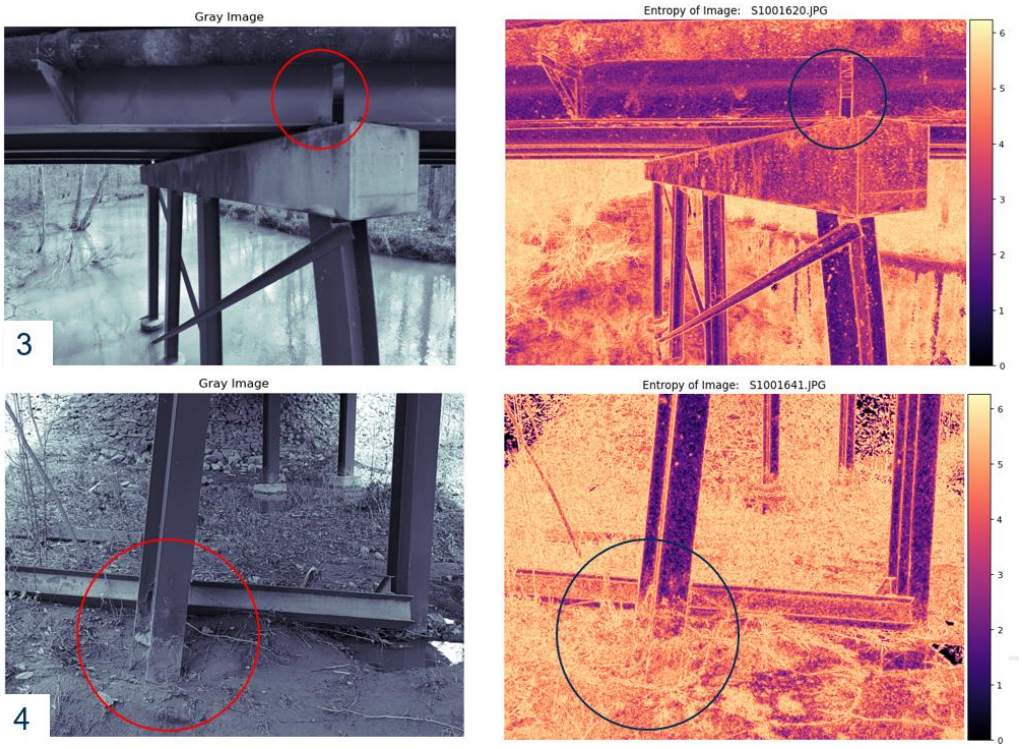

Figure 12. Photographs. Entropy of Salacoa Creek images. Circles = corrosion.

The threshold used to segment the image was determined by visual analysis. First, a group of four test images was segmented using different entropy thresholds to see at which value the corrosion was fully separated into one cluster. These thresholds ranged from 10 to 100 percent of the maximum entropy value in the image at intervals of 10 percent. After visual evaluation, it was determined that 65 percent of the maximum entropy value was the best corrosion segmentation threshold. The images were then separated into regions that were above this threshold and below it. Figure 13 shows the results of texture thresholding on four images from Bear Creek and Salacoa Creek.

Figure 13. Photographs. Results of entropy thresholding. Circles = corrosion.

As visible in figure 13, the amount of corrosion that has been completely segmented to the "textured" or "corrosion" segment of the image varies. In images 2 and 4 of figure 13, the corrosion on the bridge is almost completely segmented above the entropy threshold. However, in images 1 and 3, part of the corrosion has been falsely identified as background. In addition, it is important to note that the majority of the photograph that is labeled as "textured" or "corrosion" is background foliage. This result is unsurprising because the entropy is highest in the photographs where foliage is present. However, this results in a large number of false positives that make the inferences using this method alone inadequate for corrosion detection.

When using edge detection operators, the results were similar. More edges are present where there is foliage rather than where there is corrosion, so there are many false positives that make this method inadequate for corrosion detection. To reduce the number of false positives, the

background was removed using a deep learning algorithm, as discussed in chapter 3. Then, the entropy was used to segment corrosion in the images. The results of entropy thresholding with the background removed are shown in figure 14.

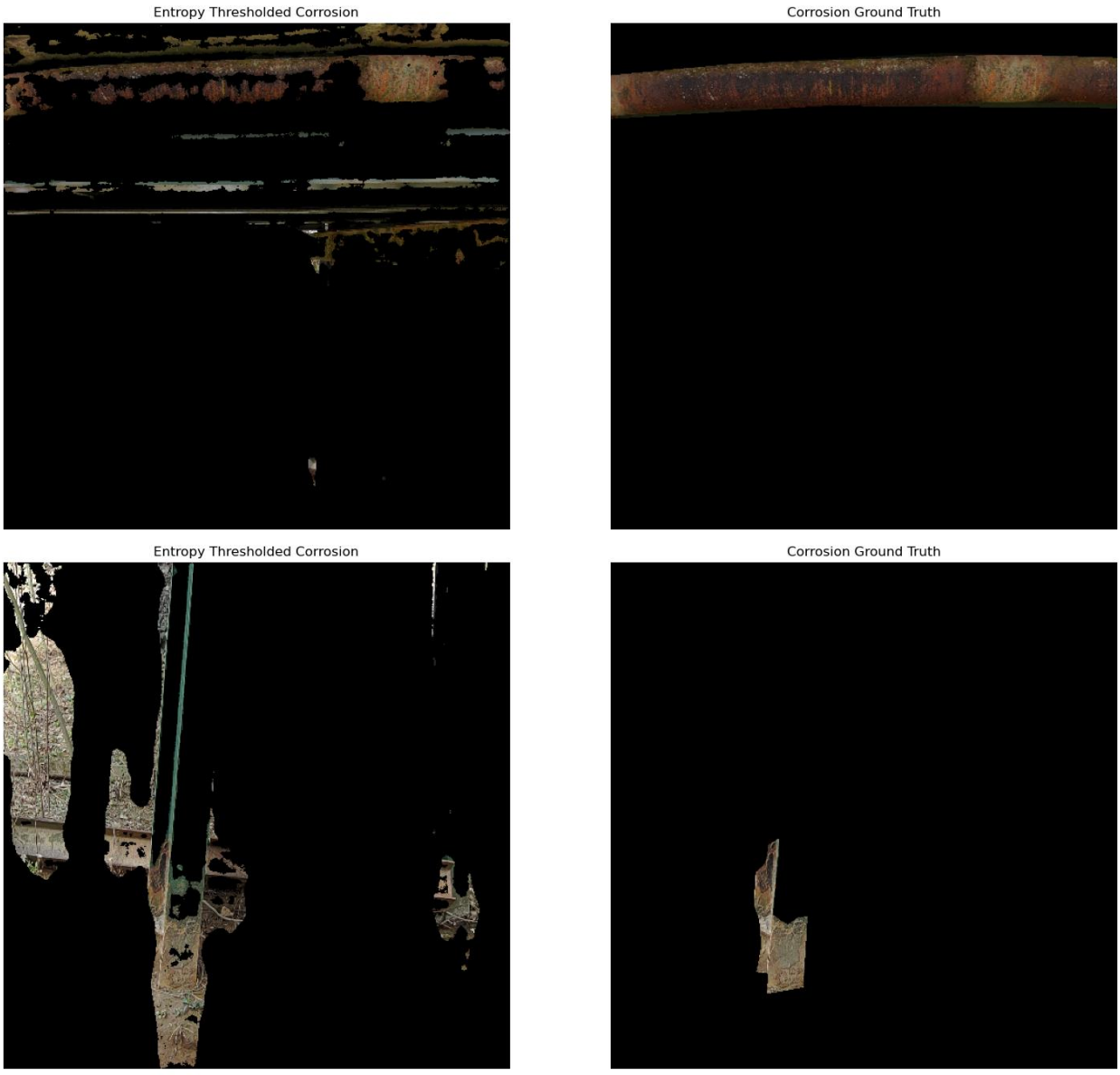

Figure 14. Photographs. Results of entropy thresholding on images with background removed using deep learning.

As shown in figure 14, there are fewer false positives on the images with the background removed than on the images with the background intact. However, it is important to note that most sections of the background that were incorrectly not removed were labeled as corrosion in this method. This highlights the potential use and advantages of an accurate background removal algorithm for corrosion segmentation on images taken in Georgia.

This methodology (i.e., automated background removal and texture thesholding) worked with high accuracy (90.1 percent) and mIoU (88.6 percent), but other performance metrics such as precision and recall were low. This is because the amount of corrosion in the images is small, so when the negative space is inferenced accurately, the overall accuracy is very high even if the corrosion is not inferenced well. Precision measures how many of the inferences are on target, recall measures how much of the target was inferenced, F1 score is the harmonic mean of precision and recall, accuracy refers to the number of pixels that were correctly inferenced, and mIoU measures the similarity of the inference and ground truth. The results are summarized in table 1.

Table 1. Performance of texture thresholding with background removed using deep learning.

Performance Metric	Value
Precision	0.0702
Recall	0.3324
F1 Score	0.0979
Accuracy	0.9005
mIoU	0.8860

To test the results of texture thresholding in more ideal conditions, this methodology was tested on images with the background removed using the image labels. This means that the background

was removed with no error. Figure 15 shows the results of texture thresholding when the background is removed using the ground truth labels.

Figure 15. Photographs. Results of entropy thresholding on images with background removed using the labels.

As shown in figure 15, the results are improved to varying degrees when the background is removed with the ground truth labels. In the top photograph, the corrosion inferences are very

similar because the deep learning algorithm removed the background with low error. However, in the bottom photograph, the deep learning algorithm did not perform as well; however, the corrosion segmentation results are much improved from when the background is removed using the deep learning algorithm. Although the accuracy did not increase, the recall increased from 0.3324 to 0.4887, which is an increase of 47.0 percent. The results of texture thresholding with perfect background removal are shown in table 2.

Table 2. Performance of texture thresholding with background removed using the ground truth.

Performance Metric	Value
Precision	0.0818
Recall	0.4887
F1 Score	0.1209
Accuracy	0.8756
mIoU	0.8613

Lastly, texture thresholding was tested on images with histogram equalization. Histogram equalization, as described in chapter 3, is used to enhance the difference of gray values in the image, which may improve the performance of texture methods. Histogram equalization was tested on images with the background removed using the ground truth, and the results are shown in table 3.

Table 3. Performance of texture thresholding with histogram equalization and background removed using the ground truth.

Performance Metric	Value
Precision	0.0818
Recall	0.5050
F1 Score	0.1236
Accuracy	0.8527
mIoU	0.8390

As shown in table 3, the precision remained the same and the recall increased slightly when the histograms of the images were equalized. The accuracy and mIoU decreased slightly, which is surprising, especially when inspecting the inferences visually. In the example images, it appears that more of the image was correctly inferenced. However, when inspecting images visually, it is difficult to determine the density of pixels in a region. For instance, a three-pixel by three-pixel square with four colored pixels and five black pixels will look similar to a three-pixel by three-pixel square with nine colored pixels, depending on the image quality. In the ground truth background scenario, there might be more "hidden" unclassified pixels, which appear black in the figures and would go unnoticed by the human eye. In these cases, these pixels might have been incorrectly classified in the ground truth scenario and not the background scenario because the texture statistics in the image changed slightly.

Although the accuracy and mIoU decreased, the change was too small to significantly impact the performance of the algorithm. On the other hand, the increase in recall from 0.3324 to 0.5050 suggests that these changes improved the methodology. The recall is arguably the most important performance metric because it tells researchers how much of the corrosion is correctly classified. If the recall is low, more corrosion will go unnoticed during bridge inspections.

COLOR THRESHOLDING

Because corrosion has a distinctive red-brown color, color thresholding was used to segment corrosion in the images. Color thresholding was tested in three color spaces: Red-green-blue (RGB), hue-saturation value (HSV), and lightness-a-b (L*ab). The RGB color space is the standard color space for digital images and has different color values stacked together to create an image. The HSV color space stacks the pixel values for hue, saturation, and value, or intensity, instead. According to previous studies, corrosion was segmented with better accuracy in HSV images than in RGB images.[11,12] Lastly, the L*ab color space separates the light values from the red-green (a) and blue-yellow (b) values. This color space was tested to see if removing the light from the images improves the performance of corrosion segmentation.

To segment the images, the 25^{th} and 75^{th} percentiles of each layer were found, and then the distance between these percentiles and each pixel was calculated. If the values of a pixel were closer to the 25^{th} percentile, that pixel was classified into the first segment, and if the values were closer to the 75^{th} percentile, the pixel was classified into the second segment. This thresholding method was used to account for the fact that each pixel had multiple values, rather than in the texture scenario, where each pixel only had one.

In the RGB color space, color thresholding did not appear to segment the corrosion in the image but separated the light spaces from the shadows. As shown in figure 16, Cluster 1 is generally where the image was darker, and Cluster 2 is where there was more light. The visible patch of corrosion in the photograph is also split between the two clusters rather than being fully segmented into one or the other. Therefore, based on visual evaluation, this color thresholding

method was deemed inadequate to segment corrosion. This method was not evaluated numerically.

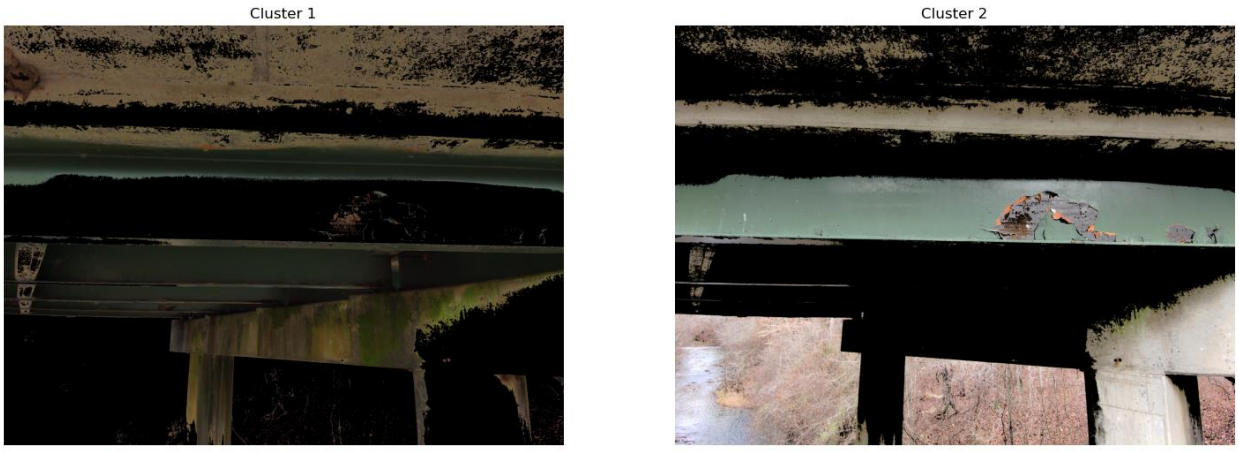

Figure 16. Photographs. Results of color thresholding in the RGB color space.

To follow the studies conducted by Petricca et al.[11] and Bondada et al.,[12] the HSV photographs were segmented using only the hue and saturation layers. Theoretically, removing the value layer from the photograph would reduce the impact of lighting. Although the lighting is not as impactful as it is for the RGB color space, the image seems to be segmented based on gray areas and colorful areas, as shown in figure 17. The large patch of corrosion present on the bottom of the pile has been split into both of the clusters, so this thresholding method was deemed inadequate to segment corrosion as well.

Figure 17. Photographs. Results of color thresholding in the HSV color space.

In the L*ab color space, the lightness layer was removed to reduce the impact of lighting. Therefore, the image was only segmented based on the pixel values of red-green and blue-yellow. The results of color thresholding in the L*ab color space are shown in figure 18. From the results of this study, this color space performed the best. The clusters appear to be separated by color rather than lightness or saturation, and the corroded patches are more fully segmented into one cluster rather than being split between both of them. Therefore, thresholding using this color space was evaluated further.

Figure 18. Photographs. Results of color thresholding in the L*ab color space.

As shown in figure 18, segmentation using this method resulted in many false positives in the background. Therefore, this method was tested on images with the background removed using deep learning in an attempt to reduce the false positives. The results are shown in figure 19. The corroded segment is shown on the left and the rest of the image is shown on the right.

As with texture thresholding, color thresholding resulted in performance with high accuracy but low precision and recall (table 4). Again, this is possible because the majority of the image represents uncorroded areas. Color thresholding was also tested on images with the background removed using the ground truth labels to see how it would perform in a perfect scenario. The results are shown in figure 20.

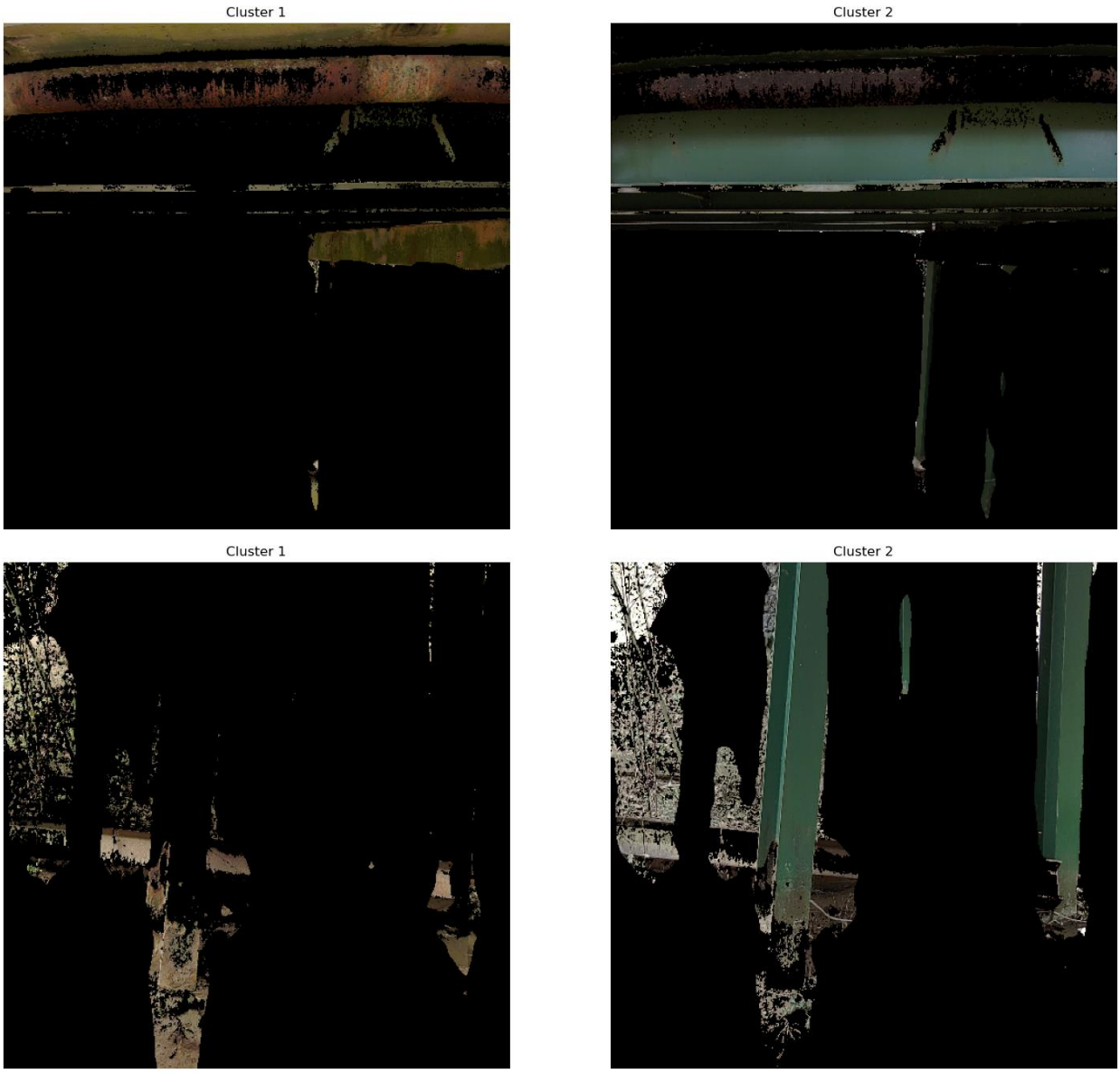

Figure 19. Photographs. Results of color thresholding on images with background removed using deep learning.

Table 4. Performance of color thresholding with background removed using deep learning.

Performance Metric	Value
Precision	0.0566
Recall	0.1571
F1 Score	0.0771
Accuracy	0.9099
mIoU	0.8954

Figure 20. Photographs. Results of color thresholding on images with background removed using the ground truth.

The performance improved slightly when the background was removed with no error. These results are summarized in table 5.

Table 5. Performance of color thresholding with background removed using the ground truth.

Performance Metric	Value
Precision	0.0564
Recall	0.2538
F1 Score	0.0817
Accuracy	0.8874
mIoU	0.8731

The only metric that increased significantly when the background was removed without error was the recall, which increased by 61.5 percent. The other metrics remained about the same, with the accuracy and mIoU decreasing slightly, which is similar to the results of texture thresholding.

K-MEANS ALGORITHM

The K-means algorithm is a standard unsupervised machine learning algorithm that is used to separate data into clusters.[13] First, cluster centers are randomly initiated. Then, each pixel is assigned to a cluster based on distance. The centers are updated to be the average value of the current member pixels, and the pixels are reassigned to the clusters that are now closest. This process is continued until the cluster assignments do not change. Theoretically, this algorithm optimizes the distance between clusters and is potentially more accurate than simple thresholding.

K-means was conducted with texture and color features with the background removed. First, this method was tested on images with the background removed using deep learning. Then, it was tested on images with the background removed using the ground truth. The results of the images with the background removed using deep learning are shown in figure 21.

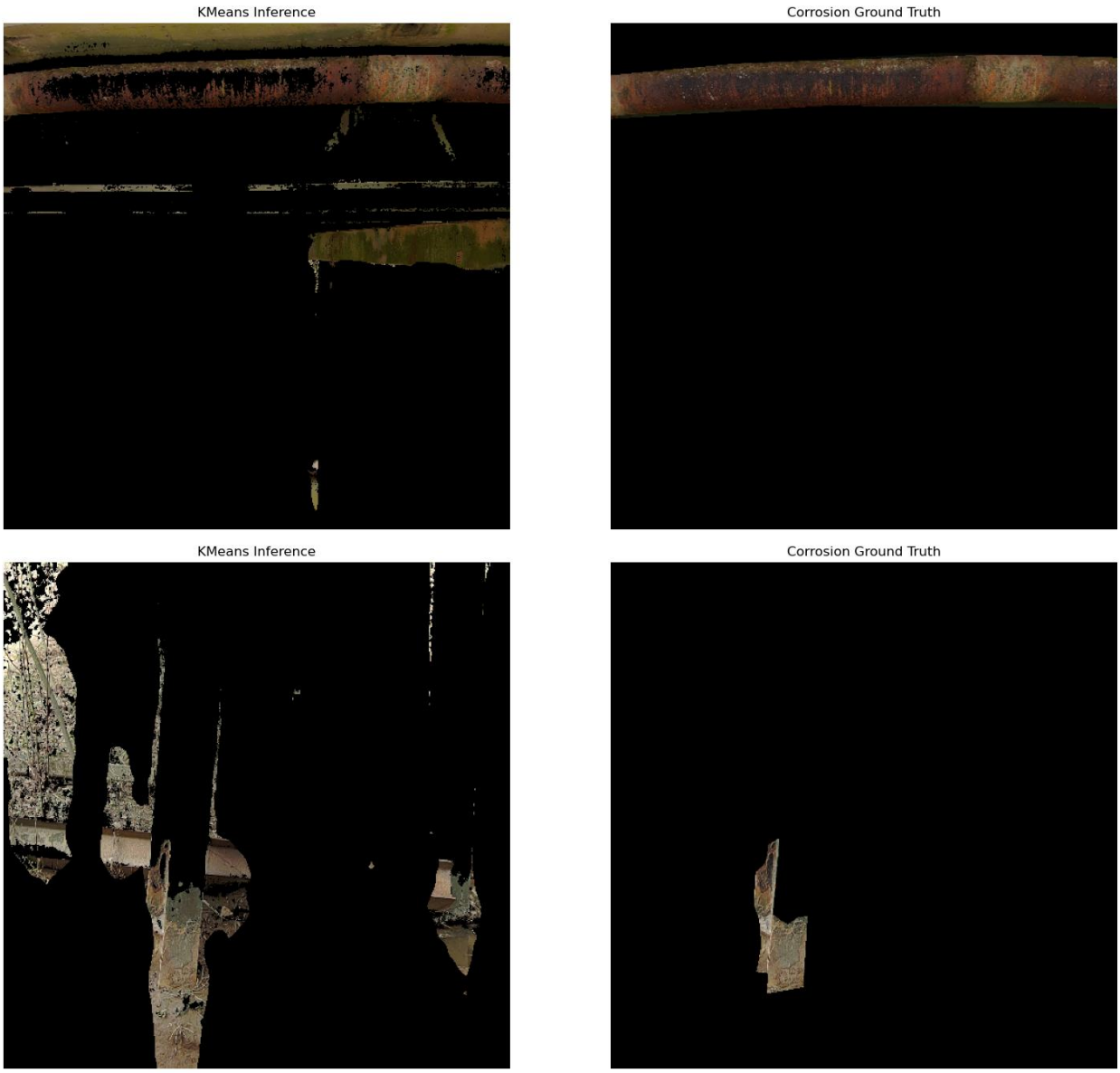

Figure 21. Photographs. Results of K-means on images with background removed using deep learning.

The performance metrics are summarized in table 6.

Table 6. Performance of K-means with background removed using deep learning.

Performance Metric	Value
Precision	0.0592
Recall	0.5129
F1 Score	0.0951
Accuracy	0.7976
mIoU	0.7834

The methodology performs with good accuracy and mIoU, but it has low precision and recall. The low recall is especially problematic because it suggests that corroded regions on the bridge could go unnoticed if this methodology were used in a bridge inspection.

Figure 22 shows the results of K-means when the background is removed using the ground truth labels.

Figure 22. Photographs. Results of K-means on images with background removed using the ground truth.

The performance metrics are summarized in table 7.

Table 7. Performance of K-means with background removed using the ground truth.

Performance Metric	Value
Precision	0.0612
Recall	0.7772
F1 Score	0.1037
Accuracy	0.7244
mIoU	0.7167

The performance of this methodology improves to varying degrees when the background is removed without error. The recall increases significantly when the background is removed perfectly, but the precision, accuracy, and mIoU decrease.

It is important to note that the K-means methodology works better on images with larger portions of corrosion. It does not work on images where there is no corrosion present because it will always segment a portion of the image as corroded. This phenomenon is shown in figure 23. As seen in the ground truth, a small amount of corrosion is present in the image, but the K-means algorithm falsely detects more regions of the image as corroded.

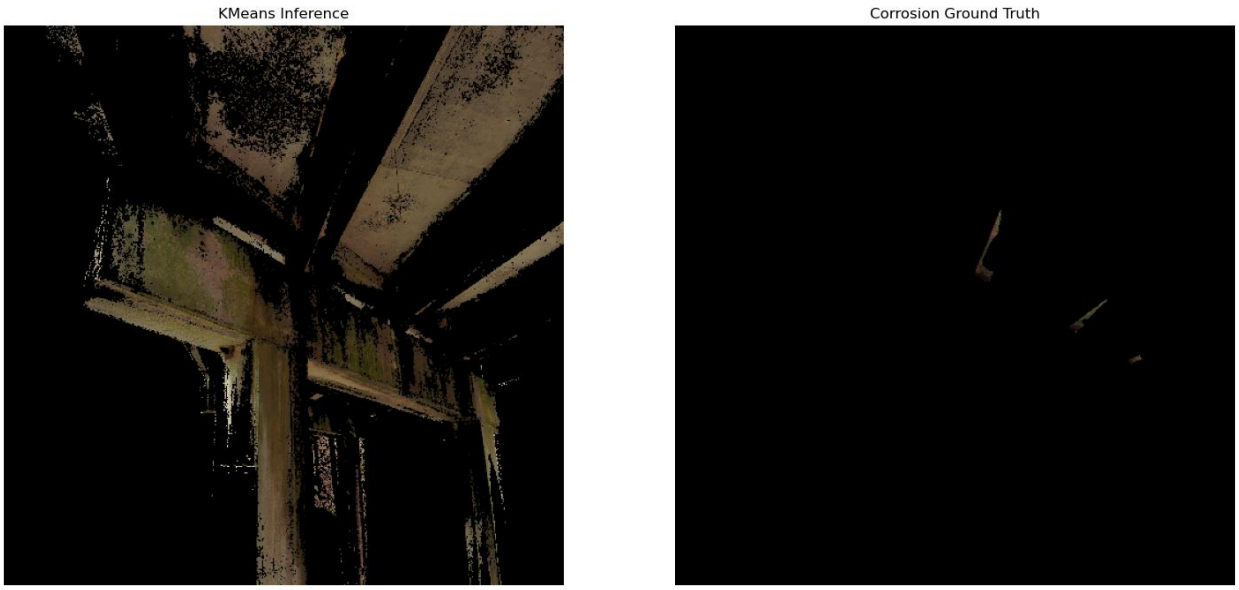

Figure 23. Photographs. K-means segmentation on an image with a small amount of corrosion.

DEEP LEARNING

Deep learning, a branch of machine learning that requires more computing power and less human intervention, has seen significant advancements in recent years. A common application of deep learning is convolutional neural networks (CNNs) for image classification and recognition.[14] The biggest advantage of CNNs is that feature extraction is included in the algorithm; as long as there are enough data samples, the feature extraction is automatically done by the convolution layers. Therefore, CNNs learn the appropriate classification features that otherwise must be hand-engineered in traditional algorithms. For corrosion evaluation, these features are color and texture, which were used to segment corrosion in the previous sections. The elimination of the need for prior knowledge and human effort, in addition to increased computing power, has led to a boom in research into CNNs.[15]

Pre-Trained Model: SpotRust

First, a pre-trained algorithm that was recently developed[16] was tested to segment corrosion on images of bridges in Georgia. This model was trained on images of pipes and performs with a reported F1 score of 0.84. Although these images were collected using drones, the corrosion takes up a larger portion of the image than in the images collected for the bridges at Bear Creek and Salacoa Creek. In addition, the background of these images is mostly sky and therefore less chaotic than the foliage in the images collected for this investigation.

When this model was used to segment corrosion on images of bridges in Georgia, the F1 score dropped to 0.1406. The performance metrics of this algorithm are summarized in table 8.

Table 8. Performance of SpotRust on images collected in Georgia.

Performance Metric	Value
Precision	0.5314
Recall	0.1660
F1 Score	0.2426
Accuracy	0.4010
mIoU	0.1606

Figure 24 shows examples of inferences made by this algorithm. The blue regions show where the algorithm predicted corrosion was present.

(a) (b)

Figure 24. Photographs. Results of SpotRust on images from (a) Bear Creek and (b) Salacoa Creek. Blue = predicted corrosion.

As shown in table 8 and figure 24, this algorithm does not infer corrosion with high recall or accuracy. This shows the importance of training a deep learning model on data that are similar to the bridge inspection data of interest.

Model Trained on GDOT Data

To improve the performance of deep learning, a pre-made architecture, Fully Convolutional Network (FCN), was trained on the images collected at Bear Creek and Salacoa Creek. Because this model would be trained specifically on data from Georgia, it was expected that it would infer corrosion on these images better than the model developed by Nash.[16] It was trained to segment regions of the bridge that were intact and regions of corrosion, which includes the staining, surface corrosion, and structural corrosion that were labeled on the images.

The training, validation, and inference were implemented in PyTorch and required about an 8.5 GB graphics processing unit (GPU). After image augmentation, the size of the dataset was 5369, and the model was trained for 50 epochs. With these hyperparameters, training took seven hours to complete. Figure 25 shows the IoU of the model throughout training, and figure 26

shows the loss of the model throughout training. The final validation loss was 0.0963 and the final validation IoU was 0.9253. Once training was complete, the model could infer the dataset almost instantaneously.

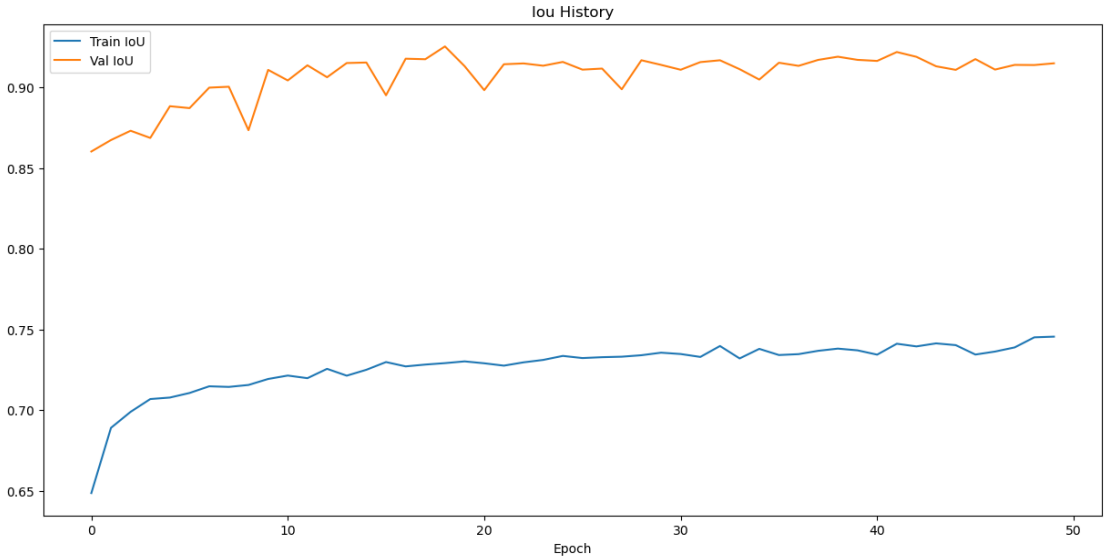

Figure 25. Graph. IoU history of FCN for corrosion segmentation.

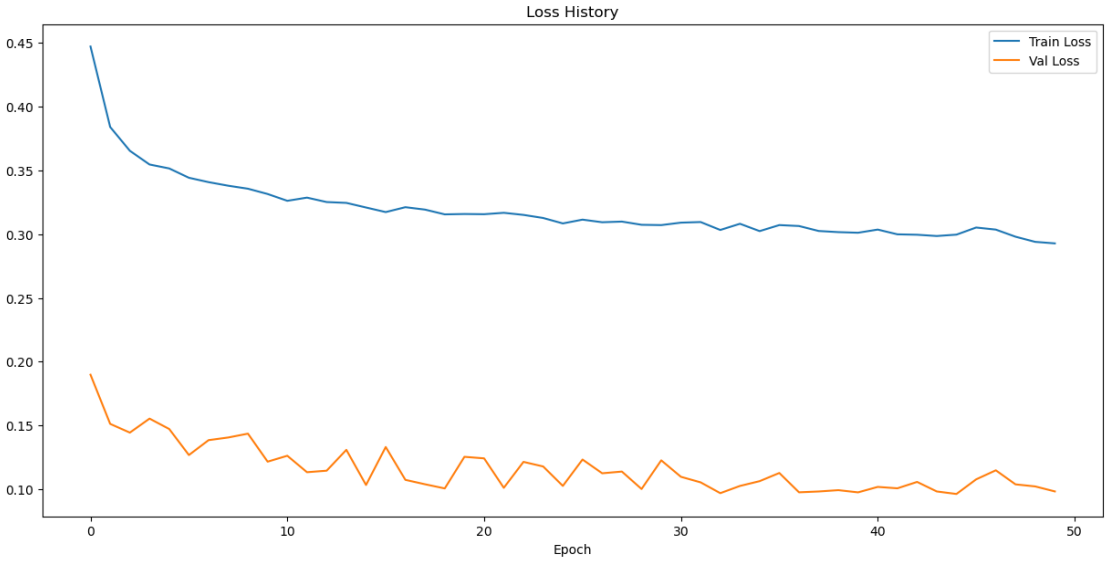

Figure 26. Graph. Loss history of FCN for corrosion segmentation.

Although the deep learning algorithm infers on the validation dataset with low loss and high IoU, it does not segment corrosion in the images at all. The segmentation is important because it indicates where corrosion is present on the bridge. Because corrosion is a small portion of the images, when the algorithm assumes there is no corrosion present on the bridge, the performance metrics seem very good. However, this methodology is unable to segment corrosion in images. The performance metrics are summarized in table 9.

Table 9. Performance of FCN trained on images from Georgia.

Performance Metric	Value
Precision	0.0000
Recall	0.0000
F1 Score	0.0000
Accuracy	0.9037
mIoU	0.9253

As shown in the table, although the mIoU is extremely high, the precision and recall are 0 because the algorithm is unable to correctly infer any corroded regions.

DISCUSSION OF CORROSION DETECTION ALGORITHMS

To be useful for bridge inspections, a corrosion segmentation technique needs to detect all corroded regions of the bridge and eliminate most uncorroded regions. These tasks are represented in the recall and mIoU, which are summarized in table 10 for all the techniques discussed.

Table 10. Summary of performance.

Algorithm	Recall	mIoU
Texture Thresholding	0.5050	0.8390
Color Thresholding	0.2538	0.8731
K-Means	0.7772	0.7167
SpotRust[16]	0.1660	0.1606
FCN	0.0000	0.9253

As shown, while some approaches work better than others, no technique as of yet works perfectly. K-means segmentation has the highest recall, but the mIoU is low. The deep learning techniques, SpotRust and FCN, perform the worst. These results show the difficulty in simply applying previous research, which showed that deep learning can segment corrosion in drone images with an accuracy above 84 percent.[3,16] However, those images do not have the same background chaos as actual bridge inspection images, such as those taken at Bear Creek and Salacoa Creek in this study that have foliage resembling the texture and color of corrosion. Also, the corrosion present on the bridges at Bear Creek and Salacoa Creek does not take up as much space in the image as the corrosion in the images from the previous studies. For these reasons, deep learning did not work well on the data collected in Georgia.

Although the deep learning methodologies did not work well on these datasets, they may perform better if they are trained on images with larger areas of corrosion in addition to the images from Bear Creek and Salacoa Creek. This will ensure that the algorithm cannot perform with low loss without segmenting the corrosion. However, retraining the algorithm will require more data collection and labeling, which are both tedious and time-consuming tasks. Therefore, it might be more useful to continue to investigate the image processing methods instead.

K-means segmentation combined with robust background removal was able to segment corrosion in these images with the highest recall. However, this method did not perform with an mIoU comparable to previous research.[9,11,12,15,16] Texture thresholding performed with the second highest recall and with an mIoU comparable to previous research. [9,11,12,15,16] The high mIoU performance results indicate that the inferences are similar to the ground truth. However, a recall of 0.7772 suggests that only 77 percent of the corrosion present on the bridge will be detected using this method. Therefore, this method must be improved before it can be used in bridge inspections. Ideally, the recall would be 1.0 for damage detection algorithms so that no damage goes unnoticed.

CHAPTER 5. CONCLUSION

This study investigated the use of drones for bridge inspections and evaluated automatic corrosion detection algorithms on data collected in Georgia. In conclusion, drones could be useful in bridge inspections because they can help inspectors reach some hard-to-view places. However, due to the large number of trees present in Georgia, many small branches around the GDOT bridges can limit where the drone can fly and how much of the bridge can be inspected. Therefore, it is recommended to conduct pre-flight planning before bridge inspections to see if thin branches or trees will be an obstacle during flight. Additionally, it is recommended to utilize the zoom feature frequently. Corrosion is more visible in the zoomed images and is detected more accurately and comprehensively in these photographs than in the standard photographs. Thus, using the zoom feature will improve the automatic corrosion detection from the UAV-aided bridge inspection. With accurate and complete automatic defect detection, drones can support inspectors' work.

The results from this research show the promise of machine learning methodologies to assist in bridge inspections. However, no technique as of yet works perfectly, and the image processing and deep learning algorithms investigated in this study need to be improved to perform well enough on Georgia data to be directly used in bridge inspections. Of the investigated algorithms, the deep learning algorithms were found to have the lowest performance on these datasets, which is contrary to previous research on corrosion detection from imagery data. Ideally, these methods would have a recall close to 1.0 to ensure that all damaged regions of the bridge are detected. The method that performed with the highest recall was K-means, with a recall of 0.7772, but it had an mIoU of 0.7167. Texture thresholding performed the second best, with a recall of 0.5050 and an mIoU of 0.8527. Alongside the promise these methodologies show in supporting bridge

inspections, this study also reveals the need to further develop them. The results presented in this study serve as a good basis to further investigate and develop image processing and machine learning methods on UAV-collected data. With further development and refinement, these methods can become useful to aid in bridge inspections.

CHAPTER 6. NEXT STEPS AND RECOMMENDATIONS

RECOMMENDATIONS FOR DRONE IMAGERY DATA COLLECTION

Based on the inspection experiences and findings from the test flights on the example bridges, we have the following recommendations for future drone flights to collect imagery data for bridge inspections.

- To reduce problems with reflectance in the images, conduct UAV-aided bridge inspections on a day that is overcast rather than partially or fully sunny. This will improve the uniformity of the lighting and improve the visibility of corrosion in the images.
- Conduct UAV-aided bridge inspections during the spring or summer when there are more leaves on the trees and surrounding vegetation so there are fewer thin branches that reduce the vision-based obstacle avoidance capabilities of the drone and limit the flyable space. This is especially true for bridges in rural areas, including the bridges investigated in Douglasville and Calhoun, but may not be as necessary for bridges in urban areas that do not have many trees nearby.
- Conduct pre-flight planning before the inspection to see if thin branches or trees will be an obstacle during flight. Inspectors can then plan for alternative ways to collect data on the angles the drone cannot reach, with the goal of collecting complete data on the bridge while they are at the site.
- Utilize the zoom feature frequently, especially when taking photographs of bridge components where corrosion is more likely, such as the bottom of piles. Corrosion is much more visible in the zoomed images and is detected more accurately and comprehensively in

these photographs than in the standard photographs, so using the zoom feature will improve the automatic corrosion detection from the UAV-aided bridge inspection.

RECOMMENDATIONS FOR DATA PROCESSING AND ANALYSIS

The findings from this study show the promise and opportunity in using drones to assist in bridge inspections. Certain methods were found to perform better than others in this initial study, with further development and refinement needed to improve the final performance metrics for use in bridge inspections.

Based on the research outcomes from investigating the varying approaches and methods for data processing and analysis, it is specifically recommended that the next phase of this work improve background removal methods and continue to investigate K-means segmentation techniques for corrosion. In all of the methods tested, the recall improved when the background was removed correctly versus based on a machine learning inference, which means that more of the corrosion was segmented when the background was removed without error. Additionally, background removal reduced the number of false positives. The background removal algorithm could be improved by training it on more varied and diverse data, such as data from other bridges with different geometries and materials. As the background removal methods improve, the chaos in the background will be less impactful for corrosion segmentation. It is also possible to remove the background by creating a three-dimensional (3D) model of the bridge using photogrammetry. The background chaos will not be present in the 3D model, which could further reduce the possibility of false positives and improve the recall.

For a given algorithm to aid in bridge inspections, it must perform with near-perfect recall and acceptable precision. Near-perfect recall is important so that damaged areas are not wrongly

eliminated or ignored. With that ability, the algorithm can act as an initial pass for bridge inspections and reduce the amount of area for the inspector to inspect without eliminating the damaged ones. Then, inspectors can review the inferences to eliminate these false positives and quantify the impact of corrosion on the bridge. A “first pass” algorithm will decrease the amount of time inspectors must spend on each asset and give them more time to quantify the impact of the damage rather than searching for it.

For these reasons, it is recommended as follow-on work to continue developing the K-means algorithm for corrosion segmentation, especially when coupled with background removal. This algorithm performed with the highest recall, and therefore eliminated the smallest amount of damaged area. K-means could also be paired with other image processing techniques, such as median filters or Gaussian filters, or be paired with other machine learning algorithms, such as naïve Bayes classifier or random forest, to see if these pairings reach near-perfect recall or improve the precision for use in bridge inspections.

REFERENCES

1. Georgia Department of Transportation (GDOT). “Bridge Maintenance Programs.” (website). Available online: https://www.dot.ga.gov/GDOT/Pages/BridgePrograms.aspx.

2. Li, G., Li, X., Zhou, J., Liu, D., and Ren, W. (2021). “Pixel-level Bridge Crack Detection Using a Deep Fusion about Recurrent Residual Convolution and Context Encoder Network.” *Measurement*, *176*, p. 109171. Available online: https://doi.org/10.1016/j.measurement.2021.109171.

3. Rahman, A., Wu, Z.Y., and Kalfarisi, R. (2021). “Semantic Deep Learning Integrated with RGB Feature-Based Rule Optimization for Facility Surface Corrosion Detection and Evaluation,” *Journal of Computing in Civil Engineering*, *35*(6), p. 04021018. Available online: https://doi.org/10.1061/%28ASCE%29CP.1943-5487.0000982.

4. Lin, W. and Yoda, T. (2017). “Chapter 1—Introduction of Bridge Engineering.” *Bridge Engineering*. Elsevier, pp. 1–30. Available online: https://doi.org/10.1016/B978-0-12-804432-2.00001-3.

5. Bianchi, E., Abbott, A.L., Tokekar, P., and Hebdon, M. (2021). “COCO-Bridge: Structural Detail Data Set for Bridge Inspections,” *Journal of Computing in Civil Engineering*, *35*(3), p. 04021003, Available online: http://dx.doi.org/10.1061/(ASCE)CP.1943-5487.0000949.

6. Kesteloo, H. (2019, Oct 1). “Skydio’s Obstacle Avoidance and Self-flying Capability Explained.” DroneDJ. Available online: https://dronedj.com/2019/10/01/skydios-obstacle-avoidance-self-flying-capability/.

7. scikit-image. (2023). “Histogram Equalization.” (website). Available online: https://scikit-image.org/docs/stable/auto_examples/color_exposure/plot_equalize.html.

8. CVAT.ai Corporation. (2022). “Computer Vision Annotation Tool (CVAT).” (website). Available online: https://github.com/opencv/cvat.

9. Medeiros, F.N.S., Ramalho, G.L.B., Bento, M.P., and Medeiros, L.C.L. (2010). “On the Evaluation of Texture and Color Features for Nondestructive Corrosion Detection.” *EURASIP Journal on Advances in Signal Processing*, *2010*(1), p. 817473. Available online: https://doi.org/10.1155/2010/817473.

10. Yuan, J., Wang, D., and Cheriyadat, A.M. (2015). “Factorization-Based Texture Segmentation,” IEEE Transactions on Image Processing, *24*(11), pp. 3488–3497. Available online: https://doi.org/10.1109/TIP.2015.2446948.

11. Petricca, L., Moss, T., Figueroa, G., and Broen, S. (2016). “Corrosion Detection Using A.I .: A Comparison of Standard Computer Vision Techniques and Deep Learning Model.” *Computer Science & Information Technology (CS & IT)*, Academy & Industry Research Collaboration Center (AIRCC), pp. 91–99. Available online: https://doi.org/10.5121/csit.2016.60608.

12. Bondada, V., Pratihar, D.K., and Kumar, C.S. (2018). “Detection and Quantitative Assessment of Corrosion on Pipelines Through Image Analysis.” *Procedia Computer Science*, *133*, pp. 804–811. Available online: https://doi.org/10.1016/j.procs.2018.07.115.

13. Roozbahani, M. “Clustering Analysis and K-Means.” (2022). Presented at the CS 7641: Machine Learning, Georgia Institute of Technology, College of Computing.

14. Yao, Y., Yang, Y., Wang, Y., and Zhao, X. (2019). “Artificial Intelligence-based Hull Structural Plate Corrosion Damage Detection and Recognition Using Convolutional Neural Network.” *Applied Ocean Research*, *90*, p. 101823. Available online: https://doi.org/10.1016/j.apor.2019.05.008.

15. Atha, D.J. and Jahanshahi, M.R. (2018). “Evaluation of Deep Learning Approaches Based on Convolutional Neural Networks for Corrosion Detection.” *Structural Health Monitoring*, *17*(5), pp. 1110–1128. Available online: https://doi.org/10.1177/1475921717737051.

16. Nash, W., Zheng, L., and Birbilis, N. (2022). “Deep Learning Corrosion Detection with Confidence.” *Npj Materials Degradation*, *6*(1), p. 26. Available online: http://dx.doi.org/10.1038/s41529-022-00232-6.