

GEORGIA DOT RESEARCH PROJECT 19-01

Final Report

**IMPROVED DISASTER MANAGEMENT
THROUGH AUTOMATED DAMAGE
ASSESSMENT USING UNMANNED AERIAL
VEHICLES (UAVs)**



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

November 2021

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No.: FHWA-GA-21-1901	2. Government Accession No.: N/A	3. Recipient's Catalog No.: N/A	
4. Title and Subtitle: Improved Disaster Management through Automated Damage Assessment using Unmanned Aerial Vehicles (UAVs)		5. Report Date: November 2021	
		6. Performing Organization Code: 6250002291	
7. Author(s): Rami J. Haddad, Ph.D., Associate Professor Maria Gonzalez Bocanegra, Undergraduate Research Assistant		8. Performing Organization Report No.: 19-01	
		10. Work Unit No.: N/A	
9. Performing Organization Name and Address: Georgia Southern University Research and Service Foundation, Inc 261 Forest Drive Statesboro, GA.30458-8005		11. Contract or Grant No.: PI# 0016892	
		13. Type of Report and Period Covered: Final; Nov. 2019 – Nov. 2021	
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		14. Sponsoring Agency Code: N/A	
		15. Supplementary Notes: Prepared in cooperation with the U.S. Department of Transportation, Federal Highway Administration.	
16. Abstract: Natural disasters cause devastating effects on transportation networks by causing significant damage and obstruction on frequently traveled roads. This report describes the design and implementation of an automated Unmanned Aerial Vehicle (UAV) based damage management using convolutional neural networks (CNNs). This system utilizes image processing and deep learning techniques to assess damage to the state's transportation system. The assessed damage are automatically geo-tagged to an ArcGIS map compatible with the Georgia Department of Transportation (GDOT) GIS standards. This UAV-based intelligent disaster management system enables the GDOT to optimize its disaster management and recovery efforts. Additionally, this system provided live streaming of the UAV's video feed to an RTMP server, enabling the first responders to assess damage. The system is composed of hardware and software components. In addition to the UAV platform, a customized application was developed using Python and MATLAB software to automate and centralize the operation of this system. The application included managing, sampling, classifying, and ArcGIS map tagging of the UAV-generated video streams. The simulation results of this system, using a library of images, have shown that the system could classify clear vs. damaged roads with an accuracy of over 99%. However, when the classification categories increased to six, damaged roads, clear roads, blocked roads, boat in roads, fallen power lines, and flooded roads, the average classification accuracy dropped to 74.1%. This was mainly due to the relatively small size of the library of disaster-related images.			
17. Keywords: Unmanned Aerial Vehicles, UAV, Convolutional Neural Networks, CNN, Natural Disaster, Deep Learning, ArcGIS		18. Distribution Statement: No Restriction	
19. Security Classification (of this report): Unclassified	20. Security Classification (of this page): Unclassified	21. No. of Pages: 99	22. Price: Free

GDOT Research Project 19-01

Final Report

IMPROVED DISASTER MANAGEMENT THROUGH AUTOMATED DAMAGE
ASSESSMENT USING UNMANNED AERIAL VEHICLES (UAVs)

By

Rami J. Haddad, Ph.D.

Associate Professor, Department of Electrical and Computer Engineering
Georgia Southern University

Georgia Southern University Research and Service Foundation, Inc.

Contract with

Georgia Department of Transportation

In cooperation with

U.S. Department of Transportation
Federal Highway Administration

November 2021

The contents of this report reflect the views of the authors, who are responsible for the facts and accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Georgia Department of Transportation or the Federal Highway Administration. This report does not constitute a standard, specification, or regulation.

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

EXECUTIVE SUMMARY.....	1
CHAPTER 1. INTRODUCTION.....	3
BACKGROUND.....	3
DISASTER MANAGEMENT.....	5
CONVOLUTIONAL NEURAL NETWORKS BACKGROUND.....	7
Pre-Trained Convolutional Neural Networks.....	9
LITERATURE REVIEW OF UAV APPLICATIONS.....	11
PROJECT OBJECTIVE.....	12
REPORT ORGANIZATION.....	13
CHAPTER 2. COMPARATIVE ANALYSIS OF UAV PLATFORMS.....	14
BACKGROUND OF UAV CLASSIFICATION.....	14
IMPORTANT UAV FEATURES AND SPECIFICATIONS.....	15
Flight Duration.....	15
Payload Capability.....	16
Software Development Integration.....	17
Ground Station Support.....	17
Ease of Customization.....	18
Reliability.....	18
UNMANNED AERIAL VEHICLE HARDWARE SYSTEMS.....	19
DJI Matrice 600 Pro Analysis.....	20
Matrice 300 RTK Analysis.....	22
DJI Phantom 4 RTK Analysis.....	27
Yuneec H520 RTK Analysis.....	29
Freefly Alta 8 Pro Analysis.....	31
DJI Mavic 2 Pro Analysis.....	33
Matrice 210 RTK V2 Analysis.....	34
HARDWARE RECOMMENDATION.....	35
CHAPTER 3. SYSTEM DESIGN AND IMPLEMENTATION.....	38
SYSTEM DESIGN OVERVIEW.....	38
SYSTEM HARDWARE.....	39
UAV Platform.....	39
Imaging System.....	39
Ground Station.....	40
SYSTEM SOFTWARE.....	41

Convolutional Neural Network Classifier.....	42
Python API and ArcGIS Software.....	42
RTMP Live Stream Server.....	44
IMAGE LIBRARY.....	46
Damaged Roads.....	46
Clear Roads.....	47
Blocked Roads.....	47
Boats in Roads.....	48
Fallen Power Lines.....	48
Flooded Roads.....	49
CHAPTER 4. GRAPHICAL USER INTERFACE APPLICATION.....	50
GUI APPLICATION DIAGRAM.....	50
LOGIN AND LOGOUT BUTTON.....	51
VIEW ARCGIS DATA BUTTON.....	54
OPEN WEBMAP WITH ITEMID BUTTON.....	59
CLASSIFY DATA BUTTON.....	61
Classify Data Only Button.....	62
Classify Data and Modify Maps Button.....	65
Use Previously Classified Data and Modify Maps Button.....	69
DOWNLOAD YOUTUBE VIDEO BUTTON.....	69
Sample Video Button.....	73
CHAPTER 5. CLASSIFICATION RESULTS AND DISCUSSION.....	75
EXPERIMENT DESCRIPTION.....	75
CLASSIFICATION QUALITY MEASURES.....	76
Classification Outcomes.....	77
Quality Measure - Recall.....	77
Quality Measure - Precision.....	78
Quality Measure - F1 Score.....	78
Quality Measure - Specificity.....	78
SIMULATION RESULTS AND DISCUSSION.....	79
Two Categories Classification.....	79
Six Categories Classification.....	80
CHAPTER 6. CONCLUSION AND RECOMMENDATIONS.....	85
CONCLUSION.....	85
RECOMMENDATIONS FOR FUTURE WORK.....	86
REFERENCES.....	87

LIST OF FIGURES

Figure 1. Chart. Average annual occurrences of billion-dollar weather-related disasters.....	4
Figure 2. Chart. Average annual cost of billion-dollar weather-related disasters (including CPI adjustment)	4
Figure 3. Map. US 2020 billion-dollar weather-related disasters.....	5
Figure 4. Illustration. Convolutional neural network architecture.....	8
Figure 5. Illustration. Architecture representation of AlexNet CNN.....	10
Figure 6. Illustration. Architecture representation of GoogLeNet CNN.....	10
Figure 7. Illustration. Architecture representation of ResNet50 CNN.....	11
Figure 8. Photo. DJI Matrice 600 Pro.....	20
Figure 9. Graph. DJI Matrice 600 flight time vs. payload.....	20
Figure 10. Photo. DJI Matrice 300 RTK.....	22
Figure 11. Graph. DJI Matrice 300 RTK flight time vs. payload.....	22
Figure 12. Photo. DJI Matrice 300 RTK camera compatibility options.....	24
Figure 13. Photo. DJI Matrice 300 RTK aircraft health management interface.....	25
Figure 14. Photo. DJI Phantom 4 RTK.....	27
Figure 15. Photo. Yuneec H520 RTK.....	29
Figure 16. Photo. Freefly Alta 8 Pro.....	31
Figure 17. Graph. Freefly Alta 8 Pro approximate flight time vs. payload.....	31
Figure 18. Photo. DJI Mavic 2 Pro.....	33
Figure 19. Photo. DJI Matrice 210 RTK V2.....	34
Figure 20. Chart. Complete system flowchart.....	38
Figure 21. Photo. Zenmuse H20 – Triple-sensor imaging system.....	40
Figure 22. Map. ArcGIS webmap with tagged disaster-damaged roads.....	43
Figure 23. Map. Information tag for each disaster-damaged road on the webmap.....	44
Figure 24. Photo. YouTube livestream link and unique key for UAV livestream.....	45
Figure 25. Photo. Disaster damaged road - Library sample image.....	46
Figure 26. Photo. Clear road - Library sample image.....	47
Figure 27. Photo. Disaster blocked road - Library sample image.....	47
Figure 28. Photo. Boat in a road - Library sample image.....	48
Figure 29. Photo. Fallen power lines - Library sample image.....	48
Figure 30. Photo. Flooded road - Library sample image.....	49
Figure 31. Chart. Complete GUI system diagram.....	50
Figure 32. Photo. GUI login window.....	51
Figure 33. Photo. GUI prompt window for ArcGIS account username.....	52
Figure 34. Photo. GUI Prompt window for ArcGIS account password.....	52
Figure 35. Photo. GUI window for login error.....	53
Figure 36. Photo. Application main window.....	54
Figure 37. Photo. Data search menu window from view ArcGIS data.....	55
Figure 38. Photo. All ArcGIS account content window	56
Figure 39. Photo. Keyword search prompt window for ArcGIS account content.....	57
Figure 40. Photo. Search results window for keyword content search.....	57

Figure 41. Photo. Title search prompt window for ArcGIS account content.....	58
Figure 42. Photo. Search results window for title content search.....	58
Figure 43. Photo. Copy ItemID to clipboard from ArcGIS account content window.....	59
Figure 44. Photo. Paste ItemID from clipboard to map ID prompt window.....	60
Figure 45. Map. ArcGIS webmap opened using the “ <i>Open Webmap with ItemID</i> ” button.....	60
Figure 46. Photo. Data classification and mapping window.....	61
Figure 47. Photo. Data selection for classification.....	62
Figure 48. Photo. Folder selection to save classification output and results.....	63
Figure 49. Photo. Saved data and results in user selected location.....	63
Figure 50. Photo. Successful classification window.....	64
Figure 51. Photo. Unsuccessful classification window.....	64
Figure 52. Photo. Webmap modification menu window after data classification.....	65
Figure 53. Photo. Webmap search window after data classification and selection of webmap modification.....	66
Figure 54. Photo. Webmap selection for modification of data.....	66
Figure 55. Photo. User selection of data to modify webmaps.....	67
Figure 56. Photo. Data successfully overwritten window.....	68
Figure 57. Photo. Overwritten map displayed in the content of the user’s GUI account	68
Figure 58. Photo. Selection of available YouTube playlists.....	70
Figure 59. Photo. Selection of available videos found in the YouTube playlist.....	70
Figure 60. Photo. Directory selection to save a downloaded video.....	71
Figure 61. Photo. Video successfully downloaded window.....	72
Figure 62. Photo. Downloaded YouTube video.....	72
Figure 63. Photo. Directory selection to save sampled frames from video.....	73
Figure 64. Photo. Output directory with the sampled video frames.....	74
Figure 65. Photo. Video successfully sampled window.....	74
Figure 66. Photos. Two category neural network classification output of disaster damaged roads.....	80
Figure 67. Graph. AlexNet confusion matrix.....	83
Figure 68. Graph. GoogLeNet confusion matrix.....	84
Figure 69. Graph. ResNet 50 confusion matrix.....	84

LIST OF TABLES

Table 1. DoD classification of UAV systems.....	14
Table 2. Summary of DJI Matrice 600 Pro features and specifications.....	21
Table 3. Summary of DJI Matrice 300 RTK features and specifications.....	26
Table 4. Summary of DJI Phantom 4 RTK features and specifications.....	28
Table 5. Summary of Yuneec H520 RTK features and specifications.....	30
Table 6. Summary of Freefly Alta 8 Pro features and specifications.....	32
Table 7. Summary of DJI Mavic 2 Pro features and specifications.....	33
Table 8. Summary of DJI Matrice 210 RTK V2 features and specifications.....	35
Table 9. Comparison between DJI Matrice 300 RTK and DJI Matrice 600 Pro	37
Table 10. AlexNet classification results.....	81
Table 11. GoogLeNet classification results.....	81
Table 12. ResNet50 classification results.....	82

LIST OF ABBREVIATIONS

AGL	above ground level
AI	artificial intelligence
AMF	action message format
API	application programming interface
AR	augmented reality
CFR	Code of Federal Regulations
CNN	convolutional neural network
CPI	Consumer Price Index
CSV	Comma-separated Values
DFOV	display field of view
DJI	Da-Jiang Innovations
DoD	Department of Defense
FAA	Federal Aviation Administration
FCC	Federal Communications Commission
FEMA	Federal Emergency Management Agency
FN	false negative
FP	false positive
GDOT	Georgia Department of Transportation
GIS	Geographic Information System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System

GPU	graphics processing unit
GUI	graphical user interface
LiDAR	Light Detection and Ranging
LRF	Laser Range Finder
LSVRC	Large-Scale Visual Recognition Challenge
MP	Megapixel
MSL	mean sea level
NOAA	National Oceanic and Atmospheric Administration
PFD	primary flight display
RF	radio frequency
RPM	rounds per minutes
RTK	real-time kinematic
RTMP	Real-Time Messaging Protocol
SDK	software development toolkit
sUAS	small unmanned aerial system
TN	true negative
TP	true positive
UAV	unmanned aerial vehicle
US	United States
XLSX	Microsoft Excel Open XML Spreadsheet

EXECUTIVE SUMMARY

With the increased rate of major weather and climate disasters occurrences in the US, the need for effective means to assess the damage and aid first responders and recovery teams has become imminent. Part of the cost incurred in such events is due to damage assessment and cleanup efforts. These efforts are usually handled manually by state and federal ground teams. Therefore, there is a growing need to develop an automated damage assessment process to streamline disaster preparedness, response, and recovery operations. Disaster management teams can optimize their recovery efforts by having access to real-time transportation network status information made possible through such a system. Additionally, such information can help federal agencies, such as Federal Emergency Management Agency (FEMA), provide the needed aid to the devastated areas.

This project aims to develop and implement an automated Unmanned Aerial Vehicle (UAV) based damage assessment system. The newly developed system utilizes image processing and deep learning techniques to classify objects and assess damage to the state's transportation system. The assessed damages are automatically geo-tagged to an ArcGIS map compatible with the Georgia Department of Transportation (GDOT) geographic information system (GIS) standards. The geo-tagged maps are used by damage assessment, response, and recovery teams to optimize their efforts, especially regarding the restoration of the state transportation system within the devastated areas. Also, the system provides the damage assessment team with a list summarizing all damages that were assessed and their geographical locations. Moreover, this system provided live streaming of the UAV's video feed to a Real-Time Messaging Protocol (RTMP) server, enabling the first responders to assess the damage.

To develop this system, a detailed comparative analysis of some of the leading commercial and industrial UAV platforms was conducted. As a result of this analysis, the final UAV platform recommendation for this project is the DJI Matrice 300 RTK. A customized graphical user interface (GUI) application was developed using Python and MATLAB software to automate and centralize the operation of the system. The application included managing, sampling, classifying, and ArcGIS map tagging of the UAV-generated video streams. This application also provided some flexibility to customize the operating settings of the system.

Due to the particular nature of this application, field tests were not viable. Therefore, the system was extensively tested using a compiled library of images captured from previous natural disasters. The image library was classified into six categories: damaged roads, clear roads, blocked roads, boats in roads, fallen power lines, and flooded roads. A total of three different convolutional neural network (CNN) (AlexNet, GoogLeNet, ResNet50) classifiers were investigated after applying transfer learning and utilizing four-fold cross-validation. AlexNet achieved the highest accuracy of 74.1%. Even though AlexNet produced the highest accuracy in this experiment, the more complex CNNs, such as GoogLeNet and ResNet50, would have provided much higher classification accuracies given larger training and testing datasets.

This project has produced the following outcomes:

- A comparative analysis of different UAV platforms.
- The design for the overall system with the hardware and software specifications detailed.
- A library of aerial images.
- A report on the classifier architecture and the performance of the classifier developed.

CHAPTER 1. INTRODUCTION

BACKGROUND

Major weather and climate disasters have been challenging our preparedness and ability to handle such events for centuries. However, in recent years, the rate of natural disasters, such as hurricanes and tropical storms, has significantly increased, primarily in the south and southeast regions of the United States (US). The US National Oceanic and Atmospheric Administration (NOAA) tracks and analyzes large-scale weather and climate disasters, focusing on those that result in over a billion dollars in economic impact. Based on NOAA's analysis of the impact of billion-dollar weather and climate disasters over the last four decades (period 1980–2020), the United States sustained 290 weather and climate disasters (with an average of 7.1 events per year) since 1980. Each event's average damage/costs reached or exceeded \$1 billion (including CPI adjustment). The total cost of these 290 events exceeded \$1.95 trillion.⁽¹⁾ To visualize the significant increase in these events over the last four decades, refer to Figures 1 and 2. Figures 1 and 2 illustrate the average annual occurrences and the average annual cost of billion-dollar weather-related disasters over periods of 10 years and compare these to the average annual occurrences and the average annual cost over the last five years (2016–2020) and also last year 2020. It is worth noting that the average annual occurrences increased from 2.9 events/year with an average cost of \$18.44B/year for the period 1980–1989 to 16.2 events/year with an average cost of \$126B/year for the period 2016–2020.⁽¹⁾

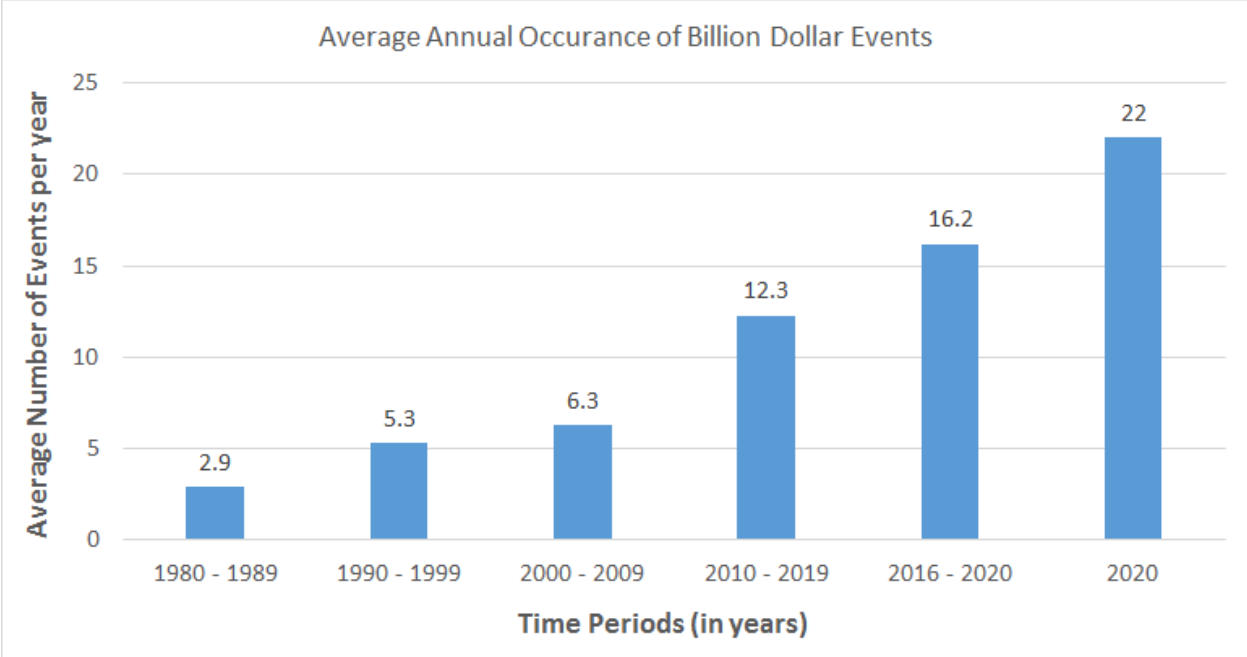


Figure 1. Chart. Average annual occurrences of billion-dollar weather-related disasters.

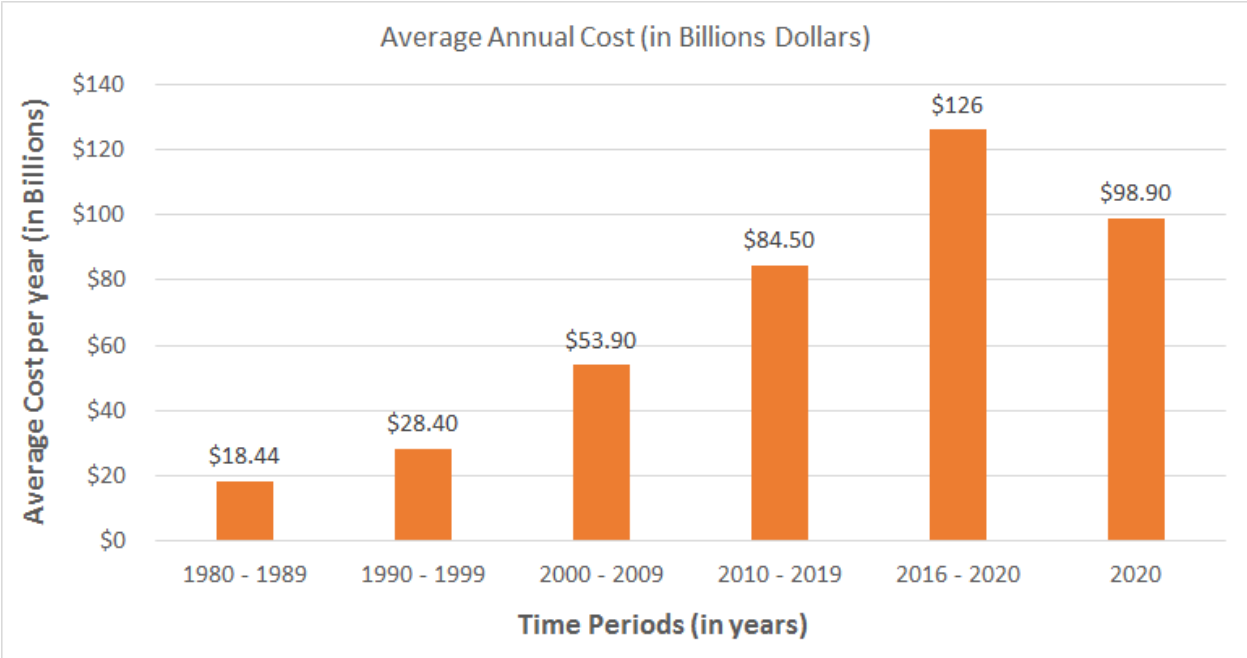


Figure 2. Chart. Average annual cost of billion-dollar weather-related disasters (including CPI adjustment).

To put things into perspective, the year 2020 set a new annual record of 22 events (with a cost of \$98.9 billion). Figure 3 illustrates the location of 22 different billion-dollar weather-related

disasters that impacted the United States during 2020.⁽¹⁾ The total cost due to weather-related disasters in 2020 was \$98.9 billion. This year was also the sixth consecutive year (2015–2020) in which ten or more billion-dollar weather and climate disaster events have impacted the United States.⁽¹⁾ Over the last decade, there were nine years with ten or more separate billion-dollar disaster events, constituting 82% of all the years with ten or more separate billion-dollar disaster events within the last 41 years (1980–2020).⁽¹⁾

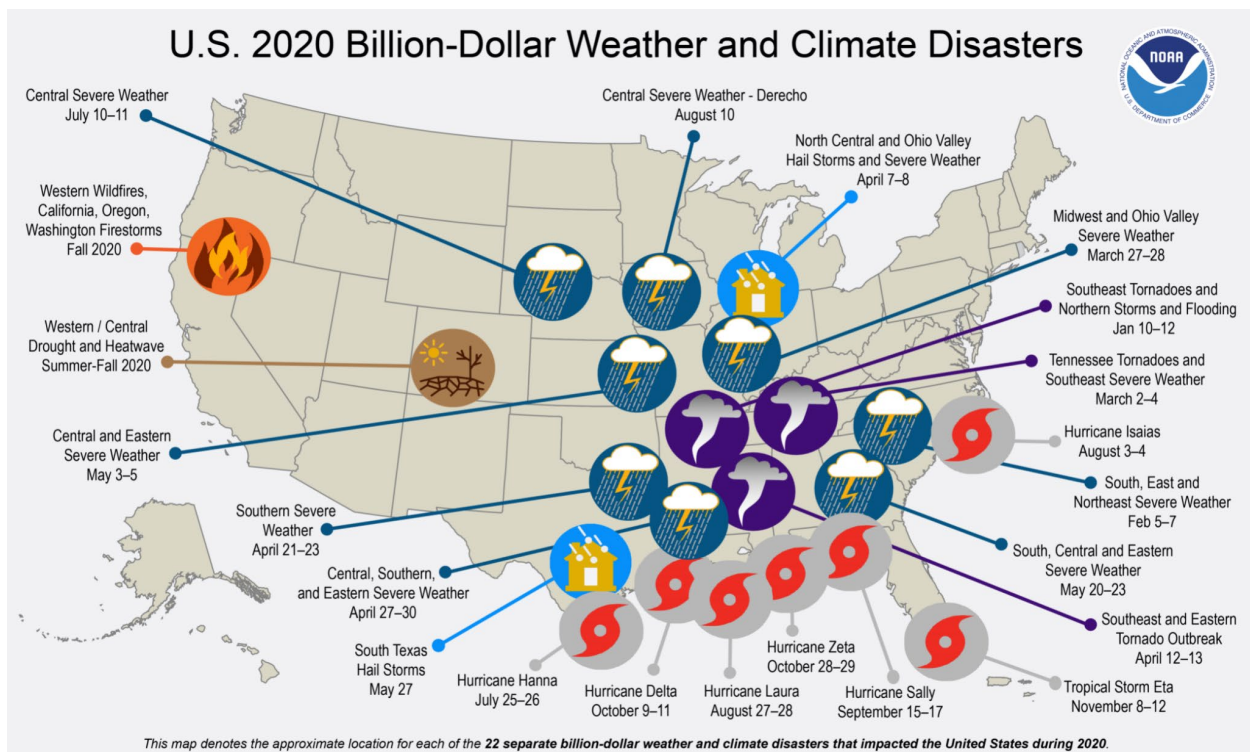


Figure 3. Map. US 2020 billion-dollar weather-related disasters.⁽¹⁾

DISASTER MANAGEMENT

Part of the incurred cost of such events is due to damage assessment and cleanup efforts. These efforts are usually handled manually by state and federal ground teams. Therefore, there is a growing need to develop an automated damage assessment process to streamline disaster preparedness, response, and recovery operations.

A system implementing Unmanned Aerial Vehicles (UAVs) and Convolutional Neural Networks (CNNs) to identify and tag disaster-damaged roads has the potential to significantly reduce the workforce needed to assess natural disaster damage and provide aid. In the year 2020, the International Committee of the Red Cross reported that the world endured a total of over three hundred natural disasters triggered by natural hazards, of which 77% were climate or weather-related.⁽²⁾ Furthermore, hurricanes, floods, tornadoes, and heatwaves had lasting effects on 1.7 billion people worldwide during the past decade.⁽²⁾ These effects include severe damage to transportation routes necessary for rapid aid administration and transit. Modern-day first responders require high-fidelity information on road damage to conduct complex recovery and aid operations on the front lines. These time-sensitive operations are affected by time delays when first responders cannot access road damage assessment data that would otherwise permit swift aid planning.⁽³⁾

The demand for assistance after a natural disaster can only be met through suitable disaster management systems. Current disaster management operations require high organizational performance from police officers, firefighters, search and rescue units, paramedics, and special casualty access teams.⁽⁴⁾ However, these methods do not address constraints such as shortage of first responders, narrow time frames, and proper distribution of aid in these situations; thus, they rely heavily on accurate geolocation data with no room for human error. Moreover, the effects of natural disasters on transportation systems produce congestion during large-scale evacuations that are detrimental to safe transit and aid transportation.⁽⁵⁾

To address such natural disaster situations, we developed a UAV system that classifies disaster-damaged roads in real-time using CNNs. This system also reports the classified damages to an online virtual map through the ArcGIS mapping software. The UAVs can capture bird's eye images of roads after natural disasters, which will later be sent to be classified. Moreover, the system can automatically tag disaster-damaged roads on an ArcGIS map to provide real-time geolocation feedback.

The proposed system would enhance first responder teams' response time by providing pertinent data (e.g., location, type of damage, timestamp, etc.) on the state of transportation routes after natural disasters. By doing so, we have expanded the application of CNNs while developing a novel and intelligent disaster management system.

CONVOLUTIONAL NEURAL NETWORKS BACKGROUND

Convolutional Neural Networks (CNNs) are flexible network configurations capable of a wide variety of image data mapping. In other words, CNNs can accurately classify input images into output categories. First proposed in the highly acclaimed 1995 paper by Yann Lecun,⁽⁶⁾ each CNN comprises three parts that help classify raw data. These parts are referred to as local receptive fields, temporal or spatial sub-sampling, and shared weights. The local receptive fields have neurons in charge of detecting and extracting prominent base features that are later combined in higher layers. In this process, shared weights, or numerical values, are implemented inside the filters used by the receptive fields to extricate the features. This creates a feature map that can be used as the starting point to guide the output. Nonetheless, the classification estimation cannot be extremely strict when identifying raw data features to produce a feature map. Therefore, spatial subsampling through activation functions is applied to reduce the resolution of feature maps and

their sensitivity to outputs produced by their current convolutional layer.⁽⁶⁾ In short, a convolutional layer in the network creates a general feature map with the essential features from the raw data. It then feeds this to the next convolutional layer as an input, which repeats the same procedure. Figure 4 exemplifies the complete architecture of a CNN, where convolutional layers can be observed creating feature maps that are then input to a pooling layer, also known as a spatial subsampling layer. The process continues until a matrix of feature layers is achieved and then fed to fully connected layers that combine these feature maps into a model.⁽⁷⁾ The output model is then classified by an activation function, in most cases the softmax “normalized exponential function” function.

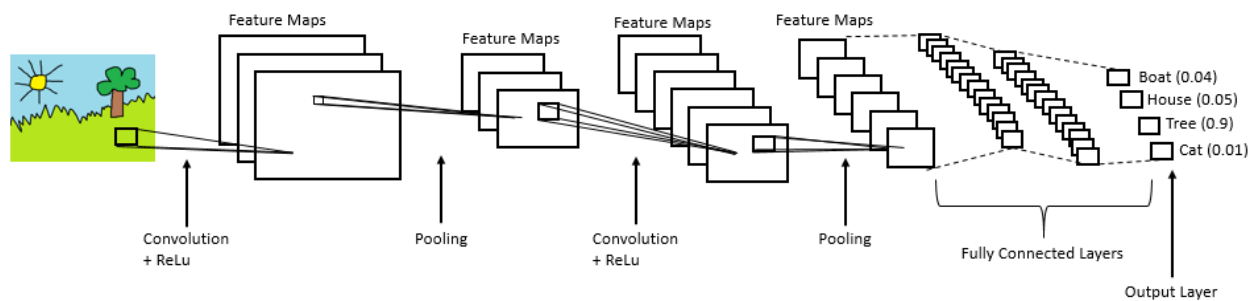


Figure 4. Illustration. Convolutional neural network architecture.⁽⁸⁾

Current image classification techniques primarily use CNNs since they are purely algorithmic and have low user dependency. The CNNs address the costs and time constraints associated with having people going through and classifying images in extensive datasets. User dependency mainly occurs during the initial training phase of the CNN. The developer will create a training dataset that will have images separated into classification categories. Thus, the network will be able to identify patterns under certain categories and properly classify future data. Artificial intelligence solutions bridge the gap between purely mathematical processes and slow human handling by creating a dynamic and trainable image classification system. Though the theories and

practical implementations of CNN-based image classifications have already been developed, there remains a need for systematic large-scale innovative implementation of these networks.

Pre-Trained Convolutional Neural Networks

Pre-trained CNNs are revolutionary for their innovative design and are capable of properly classifying many objects. The networks used in this project were pre-trained on the ImageNet dataset, dataset of 15 million high-resolution images. This significantly reduces the computational complexity in the training process by using transfer learning. Transfer learning allows pre-trained networks to modify their classification categories.⁽⁹⁾ AlexNet, GoogLeNet, and ResNet50 are some of the state-of-the-art pre-trained networks used in this project. These networks are introduced below.

AlexNet

AlexNet is a 22 layer deep CNN that classified 1.2 million high-resolution images into 1000 different classes in the ImageNet Large-Scale Visual Recognition Challenge (LSVRC) 2010 contest.⁽⁸⁾ The architecture of AlexNet is illustrated in Figure 5. Additionally, AlexNet was the first network to exploit the use of graphics processing units (GPUs) to improve performance. The AlexNet architecture consists of five convolutional layers, three max-pooling layers, two normalization layers, two fully connected layers, and one softmax layer. AlexNet's convolutional layers consist of convolutional filters and a ReLu nonlinear activation function. Its pooling layers are employed to perform max pooling. Image-wise, the network can only take a fixed size as an input, in this case, 227x227x3. Finally, AlexNet contains 60 million parameters.

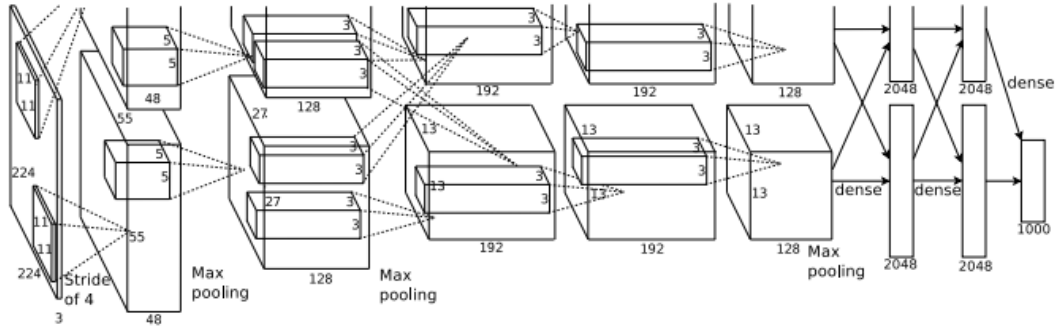


Figure 5. Illustration. Architecture representation of AlexNet CNN.⁽¹⁰⁾

GoogLeNet

GoogLeNet is a 22 layer deep CNN deployed to classify images in the ImageNet Large-Scale Visual Recognition Challenge 2014.⁽⁸⁾ The architecture of GoogleNet is illustrated in Figure 6. This neural network has been leveraged for many computer vision applications such as image classification, object detection, object classification, face recognition, etc. Architecturally, GoogLeNet accepts an input image of 224x224x3. It is worth noting that GoogLeNet does an exceptional job of reducing the input image while maintaining important spatial information through the convolutional layers. This allows the network to obtain more details from the reduced image used throughout the network. Moreover, GoogLeNet implements an Auxiliary Classifier that prevents the rise of overfitting and improves regularization.

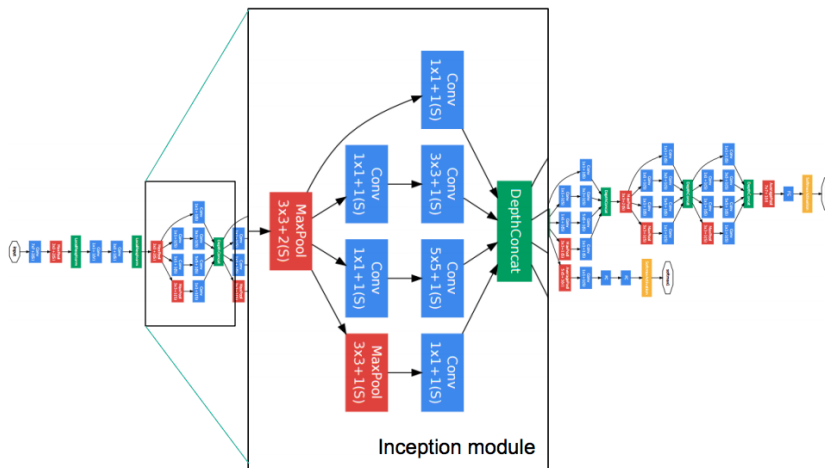


Figure 6. Illustration. Architecture representation of GoogLeNet CNN.⁽⁸⁾

ResNet50

ResNet50 is a 50 layer deep CNN trained on over a million images that implements residual learning and won first place in the ImageNet LSVRC 2015.⁽¹¹⁾ The architecture of ResNet50 is illustrated in Figure 7. The ResNet50 model consists of a total of five stages and Identity Blocks in its architecture. Each Identity Block contains three convolution layers. Fundamentally, ResNet50 is such a groundbreaking network due to its ability to train extremely deep neural networks with 150+ layers. Additionally, ResNet50 implements connection skips instead of stacking convolutional layers to fend off problems with vanishing gradient descent. Finally, the network also implements the ReLu activation layer.

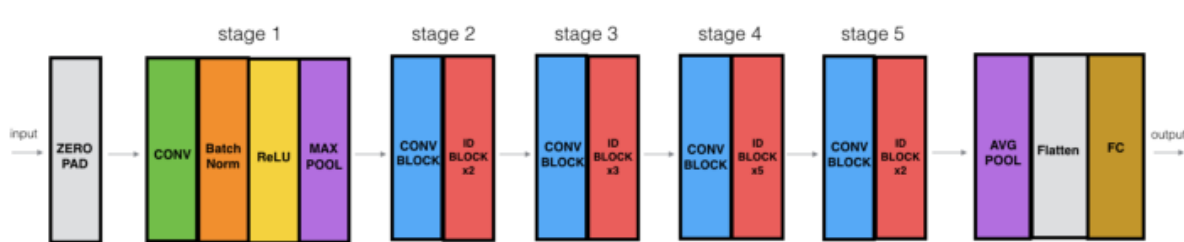


Figure 7. Illustration. Architecture representation of ResNet50 CNN.⁽¹¹⁾

LITERATURE REVIEW OF UAV APPLICATIONS

Unmanned Aerial Vehicles (UAVs) are considered an emerging technology that facilitates dynamic in-situ operations such as sensing real-time events from the air. However, minimal research has been conducted regarding the utility of UAV systems in disaster management and damage assessment systems.

There are many instances where UAVs are being utilized in transportation-related research. For example, the North Carolina Department of Transportation is investigating the use of UAV aerial photography to ensure environmental compliance during highway construction to limit the effect

of wind and water erosion of exposed soil.⁽¹²⁾ The Center for Transportation, Environment, and Community Health is investigating the utilization of UAVs with aerial photography in tracking the shoreline conditions to protect infrastructure.⁽¹³⁾ Also, a number of research projects are investigating the use of UAVs in inspecting road and railroad bridges, some of which are funded by the Mountain-Plains Consortium, Transportation Consortium of South-Central States, Transportation Research Board, and other agencies.^(14,15,16)

Further research has been conducted beyond the state transportation departments exploring the utilization of UAVs for restoring communication networks in areas devastated by natural disasters.⁽¹⁷⁾ In contrast, others have investigated the use of UAVs in traffic surveillance and monitoring.⁽¹⁸⁾ These existing studies are directly related to the scope of this proposed research. However, to the best of our knowledge, the research described herein has not been investigated before.

PROJECT OBJECTIVE

This project aims to develop an automated UAV based disaster management system. This system will utilize image processing and deep learning techniques to classify objects and assess damage to the state's transportation system. The assessed damages will be automatically geo-tagged to a map compatible with the GDOT GIS standards and ArcGIS Server/Portal maps. The geo-tagged maps will be used by the state damage assessment, response, and recovery teams to optimally facilitate their efforts, especially regarding the restoration of the state transportation system within the devastated areas. Also, the system will be able to provide the damage assessment team with a list summarizing all damages that were assessed and their geographical locations.

This proposed system will have a two-fold benefit.

1. It will provide the Georgia Department of Transportation with technology to speed up the damage assessment and recovery of the state transportation system, which will facilitate first responders' and incident management assistance teams' efforts.
2. It will also allow the Georgia Department of Transportation to identify and estimate the recovery cost for any specific event. Additionally, it will provide a means to generate reports that will assist in the recovery of GDOT assets and financial losses.

REPORT ORGANIZATION

This report is organized as follows:

- Chapter 1 is the “Introduction”, where the project background and motivation, CNNs technical background, the prior literature review, and the project objectives are discussed.
- Chapter 2, “Comparative Analysis of UAV Platforms”, contains a detailed comparative analysis of some of the leading commercial UAV systems with respect to key performance parameters. This chapter also provides UAV platform recommendations for this project.
- Chapter 3, “System Design and Implementation”, provides an overview of the system hardware and software requirements.
- Chapter 4, “Graphical User Interface Application”, details the software application features and how it centralizes the processing and mapping of the UAV data.
- Chapter 5, “Classification Results and Discussion”, contains the CNN classification simulations and results, and a discussion of the findings.
- Chapter 6, “Conclusion and Recommendations”, details the project’s conclusions and recommendations for future work.
- Finally, the report concludes with the list of references.

CHAPTER 2. COMPARATIVE ANALYSIS OF UAV PLATFORMS

BACKGROUND OF UAV CLASSIFICATION

UAVs are one of the novel technologies that offer a low-cost and straightforward means of collecting aerial imagery. Before UAVs, aerial imagery was mainly captured using manned aerial vehicles and satellites, which are not cost-effective solutions for many applications. However, UAV systems vary tremendously in size, features, and applications, so identifying the right UAV platform can sometimes be challenging. Hence, this chapter will highlight how to properly categorize UAV systems, identify key specifications, and finally assess the application viability of several commercially available UAV platforms in disaster management systems.

UAVs can be categorized using different criteria based on UAV attributes such as the type of UAV (rotorcraft vs. fixed-wing), UAV weight, payload, maximum altitude, maximum speed, etc. For example, the Department of Defense (DoD) uses five different categories based on weight, operating altitude, and airspeed to classify their UAV systems, as shown in Table 1.⁽¹⁹⁾

Table 1. DoD classification of UAV systems

UAS Category	Max Gross Takeoff Weight	Normal Operating Altitude (ft)	Airspeed
Group 1	< 20 pounds	< 1,200 above ground level (AGL)	< 100 Knots
Group 2	21-55 pounds	< 3,500 AGL	< 250 Knots
Group 3	< 1,320 pounds	< 18,000 mean sea level (MSL)	< 250 Knots
Group 4	> 1,320 pounds	< 18,000 MSL	Any Airspeed
Group 5	> 1,320 pounds	> 18,000 MSL	Any Airspeed

Meanwhile, the Federal Aviation Administration (FAA) classifies UAVs under 55 lbs. with a maximum speed of 100 mph and a maximum altitude of 400 feet as a Small Unmanned Aerial System (sUAS). Most, if not all, commercially available UAVs fall under this classification which is regulated by the FAA 14 CFR Part 107.⁽²⁰⁾

With that said, there isn't a standardized classification of UAV systems. Therefore, in this analysis, the focus will mainly be on commercially available rotorcraft UAV systems under 55 lbs. regulated by the FAA 14 CFR Part 107. However, UAV platforms vary in size, features/specifications, and price even within this category. Therefore, the following section will highlight the critical UAV features and specifications required for disaster management applications.

IMPORTANT UAV FEATURES AND SPECIFICATIONS

After a thorough review of many commercially available UAV platforms, the features and specifications that are most mentioned are flight duration, payload capability, maximum speed, maximum range, and add-on payload (such as imaging systems). However, applications such as disaster assessment require additional features and specifications such as reliability (for instance, maximum wind resistance), software development integration capabilities, ground station support, and ease of reconfiguration.

Flight Duration

Flight duration is especially important for disaster assessment UAV-based systems because of battery charging constraints. The longer the flight duration, the larger the covered area per flight is, which results in more efficient disaster assessment and recovery efforts. This will reduce the need to fly more missions and shorten the disaster management response time.

For example, according to the National Hurricane Center Tropical Cyclone Report for Hurricane Florence, “Almost every major road and highway in the southeastern portion of the state experienced flooding, with large stretches of Interstates 40 and 95, and U.S. Highway 70 is impassable for several days even after Florence had dissipated. At one point, the city of Wilmington was cut off entirely due to the floodwaters”.⁽²¹⁾ A UAV system with the longest flight duration and range will be more effective in a situation like this.

Payload Capability

The UAV system's ability to accommodate different payloads, such as high-quality variable zoom cameras, infra-red cameras, light detection and ranging (LiDAR), etc., is of the utmost importance. This is because the success of such a system is predicated on its ability to capture high-resolution aerial videos and images. Therefore, the minimum acceptable payload that the UAV system should carry is at least 1 kg – the reason being that advanced high-resolution camera systems can weigh close to 1 kg. For example, the Zenmuse H20T weighs 828 ± 5 g, while the Zenmuse H20 weighs 678 ± 5 g.⁽²²⁾ A typical payload would be around 900g; however, operating the UAV system under 75% of the maximum payload is optimal to increase the flight time and reduce the risk of overloading the UAV system. Based on this analysis, a UAV system with a payload of 1kg would be the minimum requirement to achieve the project goals, but not an optimal choice. As for an optimal payload capacity, a UAV system capable of handling 2 kg or more would be optimal to allow for any future payload improvements and to increase the system's future utility for other applications.

Software Development Integration

The scope of this research project is to develop a neural network capable of detecting inaccessible transportation routes and integrating this with an advanced UAV flight management system. The scope is not to develop a flight management system but rather to add the damage assessment capability to supplement the functionality of an existing flight management system. This is why a UAV system with a well-developed flight management system is desired. This will ensure that the flight management system is designed specifically for the UAV system of choice and is customizable. Typically, flight management systems require a low-level language capable of embedded systems programming. Many flight management systems, such as the DJI Onboard SDK, use the C++ programming language. The C++ programming language is preferred for flight management systems because it is a compiled language and hence executes significantly faster than interpreted languages such as Python or MATLAB, especially if processing capabilities are added to the UAV system. However, adding processing capabilities onboard the UAV will increase its weight, reducing its flight time; also, the artificial intelligence processing is energy-hungry, which will deplete the batteries faster and further reduce the flight time. Therefore, the project's scope is to process the videos and images in the ground station. Given that Python and MATLAB are currently among the most well-developed languages for artificial intelligence programming, those are the two languages recommended for the disaster assessment neural network classifier.

Ground Station Support

Ground station support is critical to achieving the goal of the project. Typically, UAV systems are designed to be controlled by a long-distance RF controller. These controllers are usually included with the purchase of the UAV system and have built-in autonomous flight capabilities. Some industrial/commercial UAVs are designed to be compatible with the manufacturer-supplied

controller and/or a central computer through the manufacturer-developed software development kit. The UAVs' capability of being controlled from a central computer (ground station) or interface with a ground station is an essential requirement since there is a need to post-process the videos and images and classify them using neural networks. Neural networks require significant computational power, typically using a graphics processing unit (GPU). GPUs are essential for neural network training and achieving real-time performance when deployed. The UAV system must communicate with a central computer having the optimal hardware for running the neural network in order to achieve real-time performance. The UAV system sends the videos and images it captures while the central computer handles the computations performed by the neural network.

Ease of Customization

The UAV system must be designed to be customized out of the box to avoid voiding its warranty. Typically, consumer UAVs have no expansion bays and are designed with a payload that is not meant to be changed. On the other hand, industrial/commercial UAVs typically can change payload but still lack expansion bays for different sensors and hardware, which is not optimal. The type of industrial/commercial UAV system that is recommended for this project is one that includes expansion bays compatible with multiple payloads and is designed to be customized by the user.

Reliability

Typically, UAV systems have quad-copter designs that pose reliability issues. This issue stems from a lack of redundancy in the motors. If a single motor on a quad-copter system fails, the UAV has an extremely high probability of crashing. In the event of a crash from standard flight altitude,

many important hardware components will be damaged to the point of replacement. Industrial grade UAVs improve reliability by using redundancy, such as a hex-copter design or adding safe-fail algorithms to handle failures gracefully. Hex-copters can lose at least one motor or propeller, possibly two depending on which two motors or propellers fail, and return safely to the ground. Another important reliability feature desired for any UAV system is smart power management. Smart power management systems have multiple batteries onboard that can operate the entire UAV system independently. When a single battery fails, the power management system will adjust the power consumption from the rest of the batteries to maintain operation. The combination of fail-safe algorithms and redundancies with a smart power management system provides the UAV system with robust and redundant safety mechanisms to ensure the UAV can land safely in the event of motor/propeller or battery failure.

UNMANNED AERIAL VEHICLE HARDWARE SYSTEMS

After reviewing various UAV platforms, the following list highlights the most viable candidate platforms to be used in this disaster assessment system. The UAV platforms of interest are:

- DJI Matrice 600 Pro
- Matrice 300 RTK
- DJI Phantom 4 RTK
- Yuneec H520 RTK
- Freefly Alta 8 Pro
- DJI Mavic 2 Pro
- Matrice 210 RTK V2

These UAV platforms were further analyzed and compared to finalize the UAV platform recommendations.

DJI Matrice 600 Pro Analysis



Figure 8. Photo. DJI Matrice 600 Pro

The DJI Matrice 600 Pro, illustrated in Figure 8, has a maximum range of 5 km and a variable flight duration ranging between 20 to 38 minutes depending on the payload, which is well above the minimum UAV specifications mentioned in the flight duration sections.⁽²³⁾ In Figure 9, a Flight Time vs. Payload graph provided by DJI plots the different flight times (in minutes) based on the payload applied to the UAV (in kg).

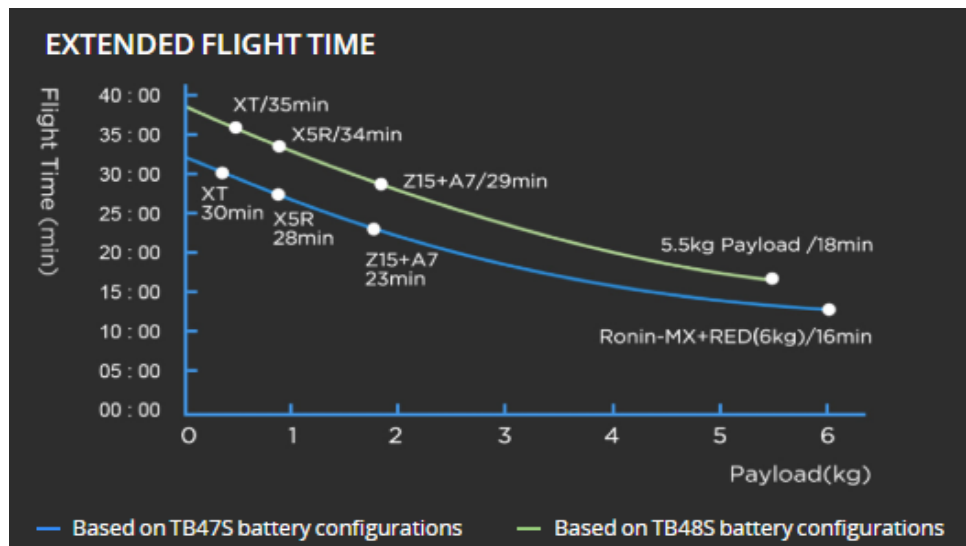


Figure 9. Graph. DJI Matrice 600 flight time vs. payload.⁽²³⁾

Referring to Figure 9, the graph shows that a payload of 2 kg would result in a flight time of 29 minutes with the DJI Matrice 600 Pro system. This UAV system is also compatible with the DJI

Onboard SDK mentioned in the Software Development Integration section. The Matrice 600 Pro implements a hexacopter design that significantly improves redundancy. In other words, if a motor stops working or a propeller malfunctions mid-flight, the UAV has a high probability of landing safely in comparison to a UAV that uses a quad-copter design. In addition, the DJI Matrice 600 Pro is compatible with one of the camera systems of choice, the Zenmuse Z30, without the need for modification. The DJI Matrice 600 Pro also utilizes a navigation and positioning system, the D-RTK GNSS. The D-RTK GNSS system would significantly improve the location accuracy when assessing damage. It is also compatible with many other camera systems and add-ons developed by DJI, making it easily reconfigured to serve different purposes. Table 2 summarizes the DJI Matrice 600 Pro features and specifications.

Table 2. Summary of DJI Matrice 600 Pro features and specifications

Maximum Range	5 km (FCC Compliant)
Maximum Hover Time	18 min (maximum payload and TB48S batteries) 38 min (without payload)
Maximum Speed	65 kph (no wind)
Maximum Payload	5.5 kg (TB48S batteries)
Maximum Wind Resistance	8 m/s
Compatible Camera Systems	DJI Gimbals: Ronin-MX, Zenmuse Z30, Zenmuse X5/X5R, Zenmuse X3, Zenmuse XT, Zenmuse Z15 Series HD: Z15-A7, Z15-BMPCC, Z15-5D III, Z15-GH4
Flight Management Systems	A3 Pro (Remote control and API compatible)
Smart Power Management	Intelligent Flight Batteries
Frame Design	Hex-copter
Number of Batteries (TB48S)	6
Battery Type	LiPo 6s
Capacity	5700 mAh
Voltage	22.8 V
Max Charging Power	180W

Matrice 300 RTK Analysis



Figure 10. Photo. DJI Matrice 300 RTK

The DJI Matrice 300 RTK, illustrated in Figure 10, is the UAV system of choice for this project. It has a maximum transmission range of 15 km and a variable flight time ranging between 55 to 31 minutes depending on the payload,⁽²⁴⁾ which is well above the minimum UAV specifications mentioned in the Flight Time section. In Figure 11, a Flight Time vs. Payload graph provided by DJI plots the different flight times (in minutes) based on the payload applied to the UAV (in kg).

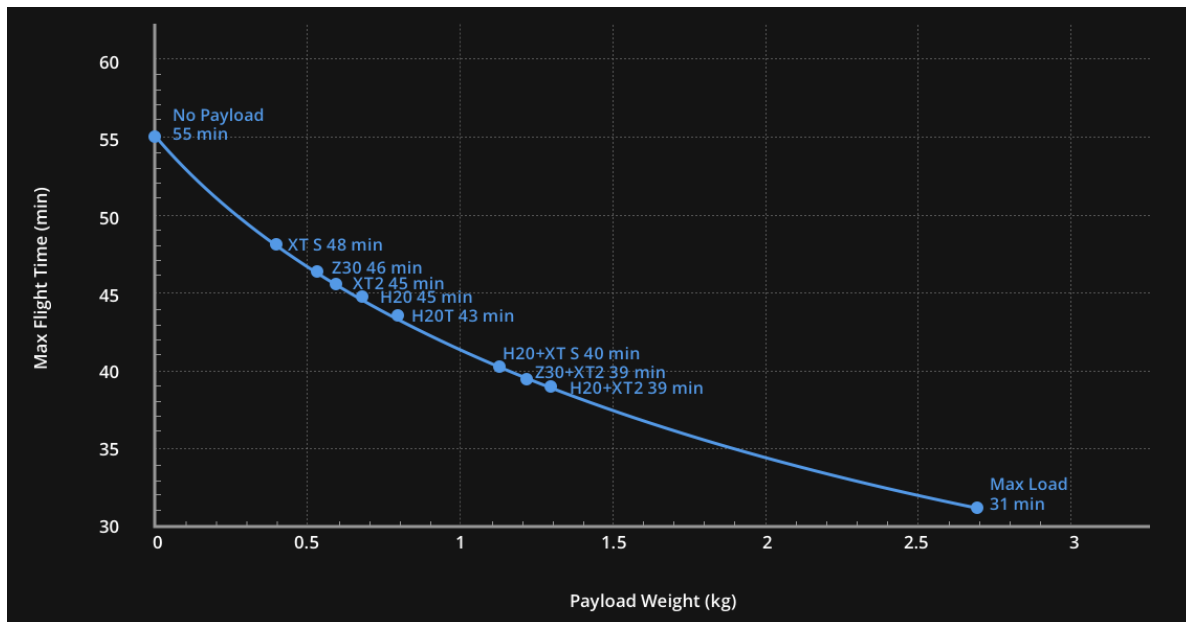


Figure 11. Graph. DJI Matrice 300 RTK flight time vs. payload.⁽²⁴⁾

Referring to Figure 11, the graph shows that a payload of 2 kg would result in a flight time of 34 minutes with the DJI Matrice 300 RTK system. The Matrice 300 RTK has a built-in advanced redundancy system that helps keep missions ongoing even in unexpected scenarios. These safety mechanisms include dual flight control system sensors, dual control signal links, dual intelligent batteries, dual transmission links, obstacle sensor system redundancies, and three-propeller emergency landing.⁽²⁴⁾

The DJI Matrice 300 RTK is compatible with almost all the high-resolution multi-sensor camera systems, such as Zenmuse H20, Zenmuse H20T, and Zenmuse Z30, without the need for modification. Figure 12 illustrates some of the Matrice 300 RTK compatible DJI camera systems. It is also compatible with many other camera systems, allowing future expansion and utilization for different applications. Furthermore, the Matrice 300 RTK has purpose-built applications such as DJI Pilot and DJI FlightHub. DJI Pilot helps the user optimize flight capability for peak performance. DJI FlightHub allows users to manage their UAV operations by supporting large organizations to scale their aerial operations.⁽²⁴⁾

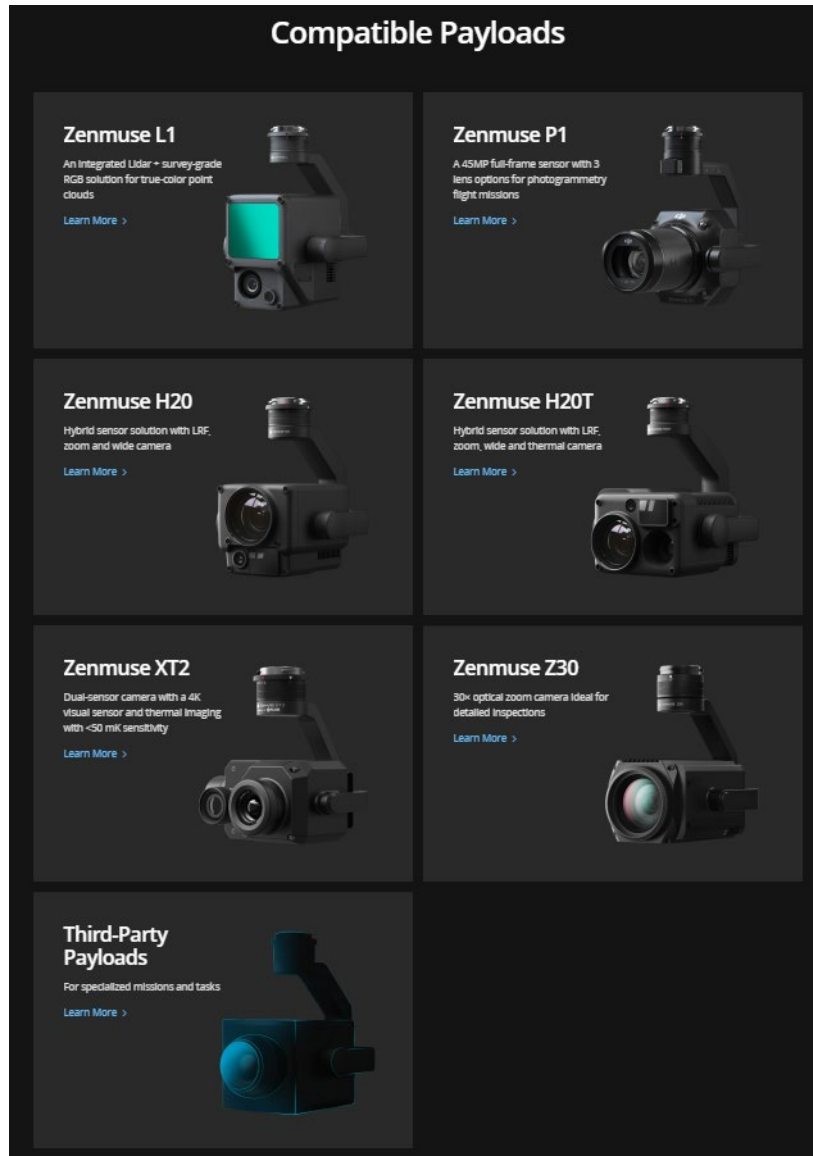


Figure 12. Photos. DJI Matrice 300 RTK camera compatibility options.⁽²⁴⁾

The DJ Matrice 300 RTK also uses a new integrated Health Management System that displays the current status of all systems, notification logs, and a preliminary troubleshooting guide, as illustrated in Figure 13. Also, this system includes the aircraft's flight logs, duration, mileage throughout its entire lifecycle, and tips on UAV care and maintenance. The Matrice 300 RTK adopts a new Primary Flight Display (PFD) that integrates flight, navigation, and obstacle information. Flight information such as aircraft attitude, altitude, velocity, wind speed, and wind direction are presented. Pilots can also view the live status of the aircraft's heading, trajectory,

PinPoint information, and home point projection more efficiently. It is also possible to visualize all nearby obstacles at once with the new obstacle map so that the pilot can be fully informed.⁽²⁴⁾



Figure 13. Photo. DJI Matrice 300 RTK aircraft health management interface.⁽²⁴⁾

The DJI 300 RTK also employs Smart PinPoint and Smart Track in its AI interface. Smart Track helps identify and follow moving subjects like people, vehicles, and boats with the Smart Track function, where auto-zoom is applied for steady tracking and viewing. The subject's dynamic location is continuously acquired and shared with another remote controller or DJI FlightHub. Smart PinPoint will mark an object in the camera or map view with a quick tap. Advanced sensor fusion algorithms will immediately calculate its coordinates, which are projected to all camera views as an AR icon. The subject's location is automatically shared with another remote controller or online platforms such as DJI FlightHub.⁽²⁴⁾

The AI component of the Matrice 300 RTK can also automate routine inspections and capture consistent results every time. Onboard AI recognizes the subject of interest and identifies it in subsequent automated missions to ensure consistent framing. The UAV can also record mission actions such as aircraft movement, gimbal orientation, photo shooting, and zoom level to create

sample mission files for future automated inspections.⁽²⁴⁾ Table 3 summarizes the DJI Matrice 300 RTK features and specifications.

Table 3. Summary of DJI Matrice 300 RTK features and specifications

Maximum Range	15 km (FCC Compliant)
Maximum Hover Time	55 min (without payload) 31 min (max payload)
Maximum Speed	82.8 kph (no wind)
Maximum Payload	2.7 kg
Maximum Takeoff Weight	9 kg
Maximum Wind Resistance	54 kph (15m/s)
Compatible Camera Systems	DJI Gimbals: Zenmuse XT2/XT S/Z30/H20/H20T
Frame Design	Quadcopter
Flight Management Systems	DJI Pilot / DJI FlightHub
Smart Power Management	Intelligent Flight Batteries - TB60
Number of Batteries (TB60)	6
Battery Type	LiPo 12S
Capacity	5935 mAh
Voltage	52.8 V
Energy	274 Wh
Ingress Protection Rating	IP45

DJI Phantom 4 RTK Analysis



Figure 14. Photo. DJI Phantom 4 RTK

A popular choice among hobbyists and professional photographers, the Phantom 4 RTK, illustrated in Figure 14, offers a reliable range and data transmission. However, this UAV system is not designed to be modified. Due to its curved design, it would be challenging to add any extra instrumentation onboard this UAV. Also, there is very little documentation on modifications and payload specifications. Overall, this UAV platform can perform the tasks required. However, it is not recommended since it lacks fail-safe mechanisms, can't handle payloads other than its onboard camera, and has a limited camera system compared to other platforms. From the research gathered on the Phantom 4 RTK UAV, this system is optimized for high-quality photographs and videos but is not optimal for this project.⁽²⁵⁾ Table 4 summarizes the DJI Phantom 4 RTK features and specifications.

Table 4. Summary of DJI Phantom 4 RTK features and specifications

Maximum Range	6.9 km
Maximum Flight Time	27 min
Maximum Speed	50 kph P - mode 58 kph A - mode
Maximum Payload	N/A (not provided by DJI and not designed for custom payloads)
Maximum Wind Resistance	10 m/s
Compatible Camera Systems	N/A (The 4k camera that comes included with the UAV is not meant to be upgraded, only replaced)
Flight Management System	DJI Mobile SDK and DJI Windows SDK
Smart Power Management	DJI Intelligent flight battery (limited to 1 onboard UAV)
Frame Design	Quadcopter
Number of Batteries (intelligent flight battery)	1
Battery Type	LiPo 4S
Capacity	5870 mAh
Voltage	15.2
Max Charging Power	160W

Yuneec H520 RTK Analysis



Figure 15. Photo. Yuneec H520 RTK

The Yuneec H520 RTK, illustrated in Figure 15, is one of the most advanced UAVs in the commercial field. It has been intensively used to map construction sites. Site supervisors employ the Yuneec H520 RTK for its built-in RTK system that allows augmented location accuracy. The Yuneec H520 RTK is an excellent UAV platform for this project. The main limitation compared to the DJI Matrice 300 RTK is in the range and the maximum flight time; however, this UAV can carry an impressive 23 kg of payload.⁽²⁶⁾ This UAV has many fail-safe mechanisms similar to the DJI Matrice 600 pro and the DJI Matrice 300 RTK. Table 5 summarizes the Yuneec H520 RTK features and specifications.

Table 5. Summary of Yuneec H520 RTK Features and Specifications

Maximum Range	1.6 km (for FCC Compliance in optimal conditions)
Maximum Flight Time	28 min (depending on payload)
Maximum Speed	61 kph (manual mode)
Maximum Payload	23 kg
Maximum Wind Resistance	Maximum wind tolerance between 10 m/s and 13 m/s
Compatible Camera Systems	CGO-3+ (3-axis gimbal 360-degree rotation), CGO-CI seven-element inspection-ready camera (cinema and vertical inspection), CGO-ET dual thermal RGB camera (For industrial maintenance, fire inspections, and search & rescue)
Flight Management System	Yuneec SDK or ST16S All-In-One Controller
Smart Power Management	Yes (can lose one motor and still fly)
Frame Design	Hexa-Copter
Battery Type	LiPo 4s
Capacity	5250 mAh
Voltage	15.2V
Max Charging Power	N/A

Freefly Alta 8 Pro Analysis



Figure 16. Photo. Freefly Alta 8 Pro

The Freefly Alta 8 Pro, illustrated in Figure 16, is designed for high-quality aerial cinematography with the ability to carry a maximum payload of 9 kg.⁽²⁷⁾ This UAV is designed to carry high-performance camera systems and has a relatively high flight time considering the payload. However, its flight time is still less than the DJI Matrice 300 RTK. The maximum range for this UAV sits at 5 km, with a maximum flight time ranging between 7 to 34 minutes depending on the battery capacity used and the actual payload.⁽²⁷⁾ Figure 17 highlights the flight time vs. payload for three different battery setups.

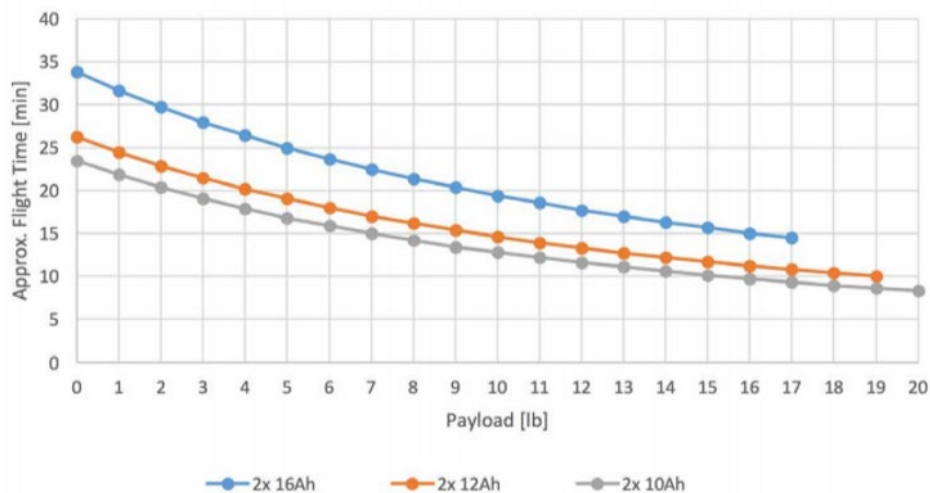


Figure 17. Graph. Freefly Alta 8 Pro approximate flight time vs. payload.⁽²⁷⁾

Additionally, the octocopter framework allows for more system stability if any of the rotors fail. The Freefly Alta 8 is a viable UAV platform for this project. Compared to the DJI Matrice 300 RTK, the main limitations are in the range and the maximum wind tolerance. Table 6 summarizes the Freefly Alta 8 Pro features and specifications.

Table 6. Summary of Freefly Alta 8 Pro features and specifications

Maximum Range	5 km
Maximum Flight Time	7 – 8 minutes (at max payload) 20 – 30 minutes (at 2 kg)
Maximum Speed	45 mph (72 kph depending on payload)**
Maximum RPM (flat rated)	3600 RPM
Maximum Payload	9 kg
Maximum Wind Resistance	8 m/s
Compatible Camera Systems	Ready-Made RC RMRC-700XVN (Recommended), Runcam Eagle 2 Pro, or similar
Flight Management System	DJI Onboard SDK (QT compatible)
Smart Power Management	Yes
Frame Design	Octocopter
Battery Type and Size	LiPo 6s (2 parallel battery packs)
Capacity	10 Ah
Voltage	22.2 V
Max Charging Power	N/A

DJI Mavic 2 Pro Analysis



Figure 18. Photo. DJI Mavic 2 Pro

The DJI Mavic 2 Pro, illustrated in Figure 18, is one of the most advanced customer-friendly UAVs in the market. The Mavic 2 Pro's design is geared to high-quality photography and hardware portability. In addition, DJI implemented various sensors on the Mavic 2 Pro's design to allow for obstacle avoidance.⁽²⁸⁾ However, the DJI Mavic 2 Pro is not recommended for this project since it cannot carry any additional payload, limiting its use of more advanced camera systems. Overall, the Mavic 2 Pro does not meet all of the requirements for this project's preferred UAV platform.

Table 7 summarizes the DJI Mavic 2 Pro features and specifications.

Table 7. Summary of DJI Mavic 2 Pro features and specifications

Maximum Range	18 km (at consistent 50 kph)
Maximum Flight Time	31 minutes (at consistent 25 kph)
Maximum Speed	72 kph (S-mode and no wind)
Maximum Payload	Not designed for custom payloads*
Maximum Wind Resistance	10 m/s
Compatible Camera Systems	Hasselblad L1D-20c Camera
Flight Management System	DJI Onboard SDK (QT compatible)
Smart Power Management	Yes
Frame Design	Quadcopter
Battery Type and Size	LiPo 4s
Capacity	3850 mAh
Voltage	15.4 V
Max Charging Power	80 W

Matrice 210 RTK V2 Analysis



Figure 19. Photo. DJI Matrice 210 RTK V2

The DJI Matrice 210 RTK V2, illustrated in Figure 19, has one of the most rugged designs on the market. It combines this design with a simple user configuration to perform industrial applications. This UAV has a built-in RTK module and also includes a GNSS mobile station. The combination of these two systems makes the DJI Matrice 210 RTK V2 capable of high precision mapping. With its intelligent flight mode features, such as obstacle avoidance, point of interest, and active track, the DJI Matrice 210 RTK V2 is one of the ‘smartest’ UAVs in the industry. This UAV is used in various applications such as firefighting, construction site mapping, telecommunication inspection, and offshore rig inspection.⁽²⁹⁾ This UAV is a good candidate platform to use for this project. Table 8 summarizes the DJI Matrice 210 RTK V2 features and specifications.

Table 8. Summary of DJI Matrice 210 RTK V2 features and specifications

Maximum Range	8 km
Maximum Flight Time	24 min (with maximum payload)
Maximum Speed	73.8 kph (45.9 mph) S-mode/A-mode 61.2 kph (38 mph) P-mode
Maximum Payload	1.2 kg
Maximum Wind Resistance	10 m/s
Compatible Camera Systems	Zenmuse XT2, Zenmuse X5S, Zenmuse X7, Zenmuse Z30, Zenmuse XT, and Third-Party Payloads
Flight Management System	DJI Onboard SDK (QT compatible)
Smart Power Management	Yes
Frame Design	Quadcopter
Battery Type and Size	LiPo 6s
Capacity	7660 mAh
Voltage	22.8V
Max Charging Power	180 W

HARDWARE RECOMMENDATION

After assessing and comparing the best commercial and industrial UAVs in the market, the DJI Matrice 600 Pro UAV was initially the recommended system for this project. The DJI Matrice 600 provides a substantial flight time of approximately 30 minutes, with the capability of handling the project's minimum required payload of 2 kg. This UAV's ability to accommodate high payloads makes this UAV system a desirable platform since it is easy to modify for future payloads and to utilize for different applications without sacrificing a substantial amount of flight time. Its hexacopter design makes the UAV extremely reliable. As mentioned before, the DJI Matrice 600 Pro can accommodate a wide variety of advanced camera systems, allowing seamless future modifications to this UAV platform. Moreover, the D-RTK GNSS software implemented in the DJI Matrice 600 Pro allows for better maneuvering and provides better accuracy.

This comparative analysis provided justifications for multiple UAV systems to choose from. However, the DJI Matrice 600 Pro met and exceeded the minimum required specifications for this

project and was recommended as the platform of choice. However, the DJI Matrice 600 Pro was discontinued during the project and was replaced with the new DJI Matrice 300 RTK. Therefore, a new comparative analysis was conducted between the new DJI Matrice 300 RTK and the DJI Matrice 600 Pro. The analysis concluded that the new DJI Matrice 300 RTK exceeds the expectations compared to the DJI Matrice 600 Pro in key areas such as wind speed tolerance, range, speed, flight time, and reliability.

Furthermore, the Matrice 300 RTK applies newer AI-based technology such as Smart Track and Live Mission Recording, enabling the user to have an automated system that returns accurate data for mission completion. Therefore, the final recommendation is to utilize the DJI Matrice 300 RTK as the UAV platform of choice for this project. The main reason why this platform was chosen is due to its impressive ability to resist wind speeds up to 54 kph or 15 m/s, which surpassed all other UAV platforms analyzed. Table 9 summarizes the one-to-one comparison between the DJI Matrice 300 RTK and the DJI Matrice 600 Pro.

Table 9. Comparison between DJI Matrice 300 RTK and DJI Matrice 600 Pro

	DJI Matrice 300 RTK	DJI Matrice 600 Pro
Maximum Range	15 km (FCC Compliant)	5 km (FCC Compliant)
Maximum Hover Time	31 min (max payload) 55 min (without payload)	18 min (maximum payload and TB48S batteries) 38 min (without payload)
Maximum Speed	82.8 kph (no wind)	65 kph (no wind)
Maximum Payload	2.7 kg	5.5 kg (TB48S batteries)
Maximum Takeoff Weight	9 kg	15.5 kg
Maximum Wind Resistance	54 kph (15 m/s)	28.8 kph (8 m/s)
Compatible Camera systems	DJI Gimbals: Zenmuse XT2/XT S/Z30/H20/H20T	DJI Gimbals: Ronin-MX, Zenmuse Z30, Zenmuse X5/X5R, Zenmuse X3, Zenmuse XT, Zenmuse Z15 Series HD: Z15-A7, Z15-BMPCC, Z15-5D III, Z15-GH4
Frame Design	Quadcopter	Hex-copter
Flight Management systems	DJI Pilot / DJI FlightHub	A3 Pro (Remote control and API compatible)
Smart Power Management	Intelligent Flight Batteries - TB60	Intelligent Flight Batteries
Number of Batteries (TB60)	6	6
Battery Type	LiPo 12S	LiPo 6s
Capacity	5935 mAh	5700 mAh
Voltage	52.8 V	22.8 V
Energy	274 Wh	99.9 Wh
Ingress Protection Rating	IP45	N/A

CHAPTER 3. SYSTEM DESIGN AND IMPLEMENTATION

SYSTEM DESIGN OVERVIEW

In this chapter, the system's hardware and software implementation used in this project are detailed. In this project, a novel and completely automated disaster assessment system was developed. This system captures high-quality geospatial videos of disaster-damaged roads while live-streaming the video feed to an RTMP server. Then a ground station samples the videos into images and classifies them. For each classified image, the geolocation information is extracted using MATLAB to obtain the longitude, latitude, and time-stamp. These geolocation references are then saved alongside the classification output to create a database of all the classified images. Once this step is completed, the images are sent through a Python-built interface to create an ArcGIS live map. The map is updated each time new data is received. A flowchart detailing each stage of the system is illustrated in Figure 20.

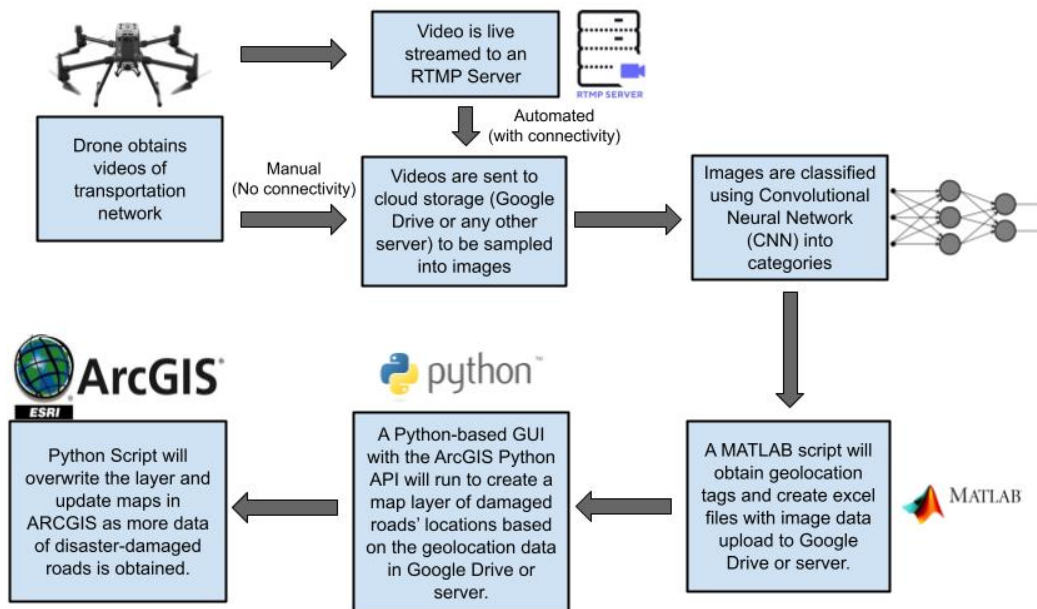


Figure 20. Chart. Complete system flowchart

SYSTEM HARDWARE

The system hardware consists of three main components, which are:

- 1) UAV platform
- 2) Imaging system (UAV add-on)
- 3) Ground station

Each of these hardware components is discussed in detail in the following subsections.

UAV Platform

To effectively capture images to feed the classifier, the UAV hardware platform of choice was initially the DJI Matrice 600 Pro. However, this UAV platform was discontinued and replaced with the newer DJI Matrice 300 RTK. After assessing and comparing the best commercial and industrial UAVs in the market, the final platform recommended was the DJI Matrice 300 RTK. This UAV provides a substantial flight time of approximately 33 minutes, with a maximum payload of 2.7 kg, exceeding the payload needed for this project and provides the most stability under windy conditions of up to 54 kph.⁽²⁴⁾

Imaging System

As for the imaging system payload, the Zenmuse H20, illustrated in Figure 21, was the add-on of choice.



Figure 21. Photo. Zenmuse H20 – Triple-sensor imaging system.⁽¹⁵⁾

The Zenmuse H20 imaging system weighs 678g and provides far more capabilities than any other regular camera. This imaging system has two cameras: the first is a wide-angle camera with a display field of view (DFOV) of 82.9° and resolution of 12 MP, while the second camera is a 20 MP Zoom camera with 23x hybrid optical zoom, up to 200x digital zoom, night mode and 4K video resolution. This imaging system also has an integrated laser rangefinder (LRF) that measures the distance to an object up to 1200 m away. This imaging system also has advanced features that further extend its utility, such as AI Spot-Check (which automates routine inspections using the camera's onboard artificial intelligence), High-Res Grid Photo, PinPoint (which allows GPS coordinate tagging of objects on the fly), Smart Track (which identifies and follows moving subjects using AI and the auto-zoom function, while continuously acquiring the subject's dynamic location).⁽²⁴⁾

Ground Station

The recommended hardware setup for the ground station is a computer that allows for high computation capacity. The computer should have a Graphics Processing Unit (GPU) to conduct a fast classification of images. GPUs accelerate the computational capabilities of a normal computer

and can run multiple processes at the same time. Since the neural network classification is developed in MATLAB, the ground station should have a recent version of MATLAB software installed. In addition to the MATLAB software, the MATLAB neural network packages for AlexNet, ResNet50, or GoogLeNet should be added.

As for the graphical user interface (GUI) application, the software needed is Python. Within Python, the GUI's foundations are PyQt5 (version 5.15.4 or latest), Tkinter (version 8.6.11), and the ArcGIS API (version 4.20). PyQt 5 is a library that implements 35 modules and high-level APIs that enable Python to be used as an application development language. Additionally, PyQt5 is the library used to create the GUI and all of its functionalities. Similar to PyQt5, Tkinter also provides GUI solutions.

A *bash* script was created to automate the installation of all the packages and dependencies necessary to run the application and the classification network for this project.

SYSTEM SOFTWARE

The system software consists of three main components, which are:

- 1) Convolutional Neural Network Classifier
- 2) Python API and ArcGIS Software
- 3) RTMP Live Stream Server

Each of these software components is discussed in detail in the following subsections.

Convolutional Neural Network Classifier

Convolutional Neural Networks were used to create the disaster assessment classifier. The CNN selection was based on the network's ability to produce the most efficient accuracy while minimizing the computational complexity. Other networks with denser architectures may produce slightly better accuracy. This performance difference might not be statistically significant compared to other less dense CNN accuracies, and it comes at the cost of more computational complexity. Three different CNN architectures – AlexNet, GoogLeNet, and ResNet50 – were evaluated in the project to find the optimal architecture for this application. These classifiers were trained to classify six different categories of damage: damaged road, clear road, blocked road, boat in the road, fallen power lines, and flooded road.

The classification performance was optimized by adjusting the training parameters (batch size, epoch number, loss functions, and learning rates). The need to structurally change the network architecture occurred when it became clear that the training parameters were no longer improving performance for classification accuracy. Due to the small training and testing dataset size, the first step was to use transfer learning techniques to repurpose pre-trained neural networks. Transfer learning improves networks' accuracy. However, due to the small size of the dataset used, cross-validation was also used to avoid overfitting. Once these methods were applied, the training parameters were reassessed.

Python API and ArcGIS Software

ArcGIS is geographical information software used to map, visualize, and analyze geospatial information. The software allows for the creation and implementation of reliable maps based on

layering information. This layering allows the creation of complex maps where data can be studied further. In this project, ArcGIS is leveraged to create maps with exact pinpoints of the locations where the UAV identified disaster-damaged roads, as illustrated in Figure 22.

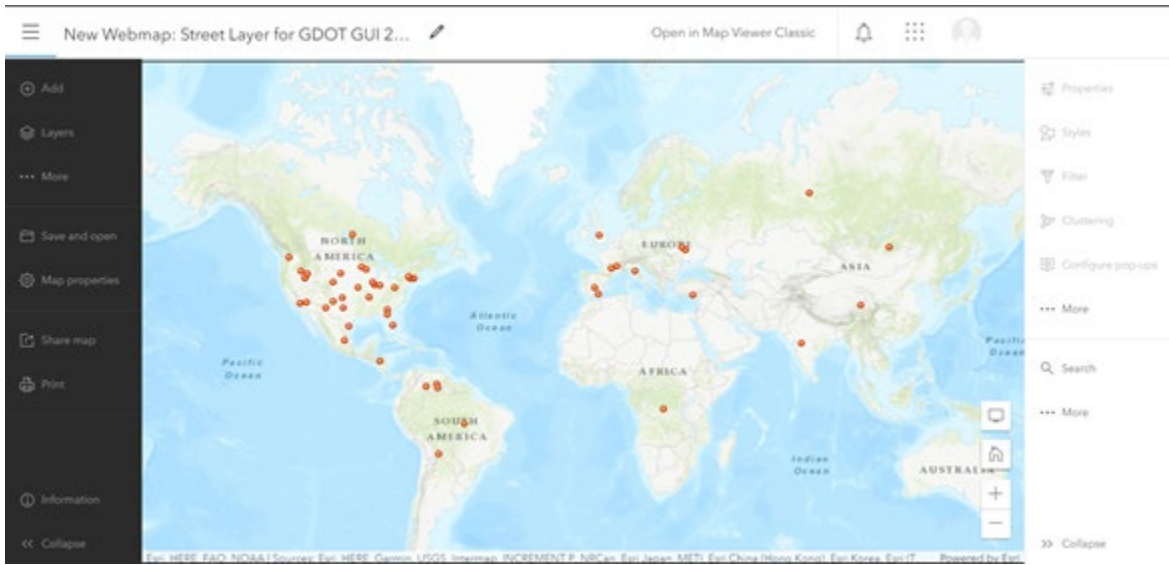


Figure 22. Map. ArcGIS web map with tagged disaster-damaged roads

Each pinpoint shows the user the latitude, longitude, and type of damage associated with a specific location. These features are illustrated in Figure 23. Furthermore, these maps can be accessed online and viewed by GDOT personnel anywhere and anytime, improving accessibility to this critical information.

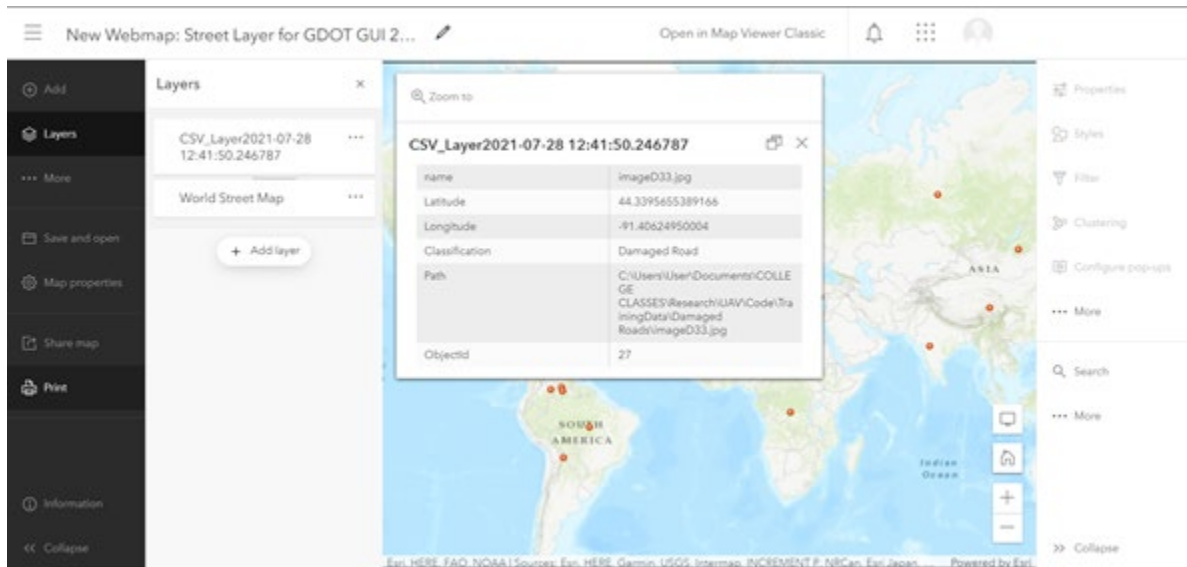


Figure 23. Map. Information tag for each disaster-damaged road on the webmap

The ArcGIS software also allows the automation of map development in their online ArcGIS platform through a Python API. The API is utilized heavily in developing the GUI to automate workflows and speed up data retrieval. The API is the backbone of the GUI since it allows for a single environment to handle the classification of data and the development, modification, and maintenance of maps.

RTMP Live Stream Server

RTMP stands for Real-Time Messaging Protocol. The server allows for high-speed transmission of video, audio, and image data from an encoder to a server. Adobe originally developed RTMP to work with Adobe Flash Player. The DJI Matrice 300 RTK system can stream a video feed using RTMP to any RTMP server. Users can use this live stream to follow the operation closely from a remote location. The RTMP implementation requires setting up an RTMP server or using pre-existing RTMP servers such as YouTube or others. In all cases, the user must enter a unique

streaming key into the UAV controller and ensure that the UAV has connectivity to allow live streaming.

Figure 24 showcases an example of the implementation of the YouTube link and unique key necessary to start a live stream with the DJI Matrice 300's controller. It is worth noting that the original video's metadata is overwritten by YouTube's metadata, therefore losing the original geolocation data. Consequently, it is necessary to use a private RTMP server for this purpose.

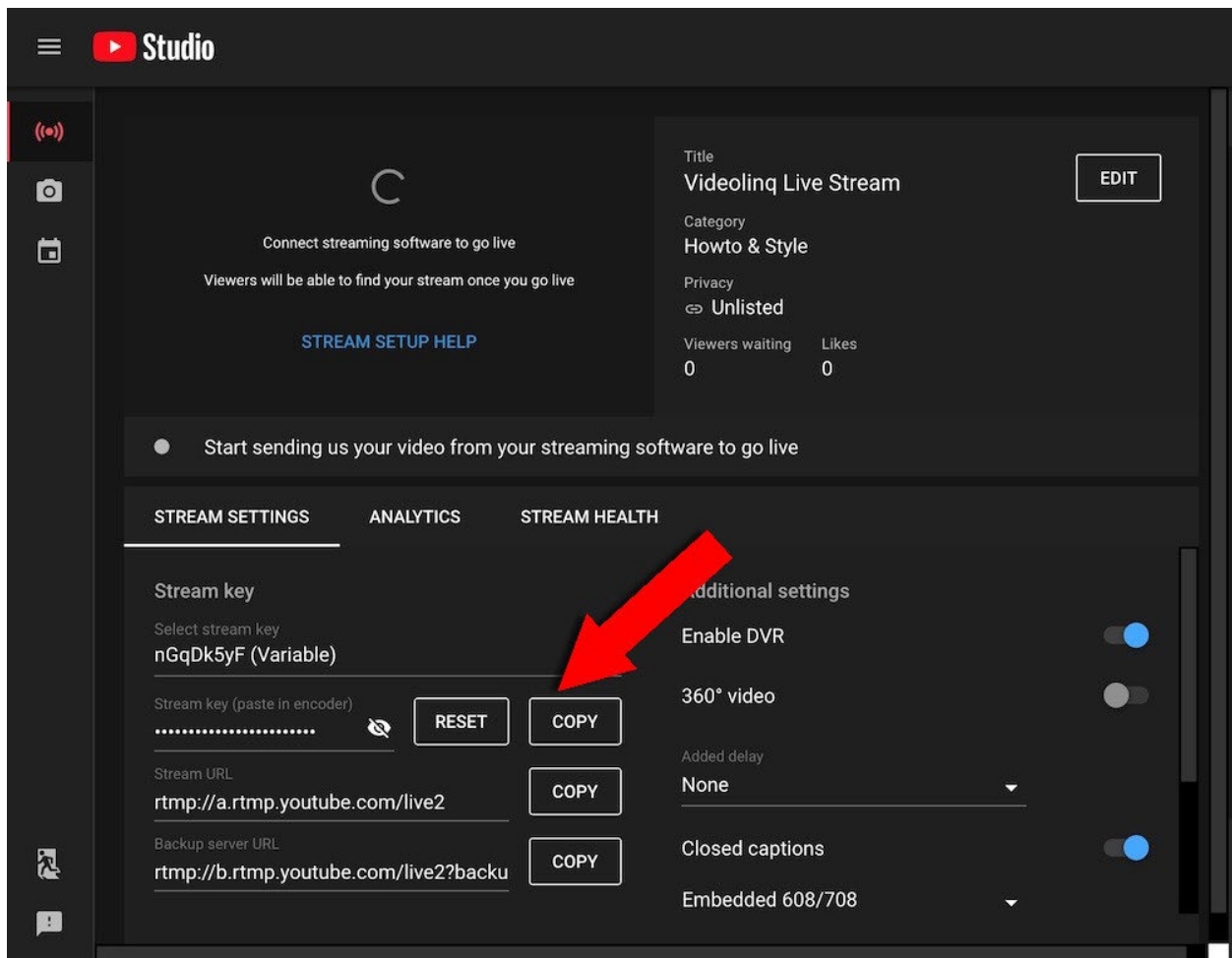


Figure 24. Photo. YouTube livestream link and unique key for UAV livestream

IMAGE LIBRARY

To develop an effective neural network classification, a large set of images representing the different categories is needed to train the neural network classifier. Therefore, a library of disaster-damaged roads was created since such a library did not exist previously. This library is composed of six different classes of damage encountered by the Georgia Department of Transportation across the state during and after natural disasters. The categories included in the library are the following: Damaged Roads, Clear Roads, Blocked Roads, Boats in Roads, Fallen Power Lines, and Flooded Roads. The images inside the library are captured from a bird's eye view or high camera angle to resemble what a flying UAV would capture in real-time. The library contains around 600 images, with each category containing 80 to 100 images.

Damaged Roads

The Damaged Roads classification is exemplified by the encountering of potholes, cracked concrete, and concrete dislocation. Figure 25 showcases a sample of a damaged road image that is fed to the classification neural network to train it.



Figure 25. Photo. Disaster damaged road – Library sample image.⁽³⁰⁾

Clear Roads

The Clear Roads classification is exemplified by the lack of any damage, obstruction, flooding, or fallen power lines. This classification focuses on the assessment of a functional road. Figure 26 showcases a clear road image sample fed to the classification neural network to train it.



Figure 26. Photo. Clear road – Library sample image.⁽³¹⁾

Blocked Roads

The Blocked Roads classification is exemplified by the obstruction of any kind to a road, except for boats. This classification focuses on the assessment of a road that does not allow proper passage and mobilization. Figure 27 showcases a blocked road image sample fed to the classification neural network to train it.



Figure 27. Photo. Disaster blocked road – Library sample image.⁽³²⁾

Boats in Roads

The Boats in Roads classification is exemplified by the obstruction of a road by a boat. Similar to Blocked Roads, this classification focuses on assessing a road that does not allow proper passage and mobilization. Figure 28 showcases a sample of an encountered boat in a road image that is fed to the classification neural network to train it.



Figure 28. Photo. Boat in a road – Library sample image.⁽³³⁾

Fallen Power Lines

The Fallen Power Lines classification is the encounter of fallen power lines on or near roads. High winds or fallen branches usually cause these fallen power lines. Figure 29 showcases a fallen power line image sample fed to the classification neural network to train it.



Figure 29. Photo. Fallen power lines – Library sample image.⁽³⁴⁾

Flooded Roads

The Flooded Roads classification is the encounter of extreme amounts of water on roads. These roads are deemed inaccessible to rescue teams due to the high water levels. Figure 30 showcases a sample of a flooded road image fed to the classification neural network to train it.



Figure 30. Photo. Flooded road – Library sample image. [\(35\)](#)

CHAPTER 4. GRAPHICAL USER INTERFACE APPLICATION

GUI APPLICATION DIAGRAM

As previously mentioned, the project incorporates a graphical user interface (GUI) developed using a Python API that automates the creation of maps and layers. The GUI application creates a centralized system in which the user has multiple options to create or modify maps and handle data collected by unmanned aerial vehicles. Figure 31 illustrates the complete GUI system diagram and its functionality.

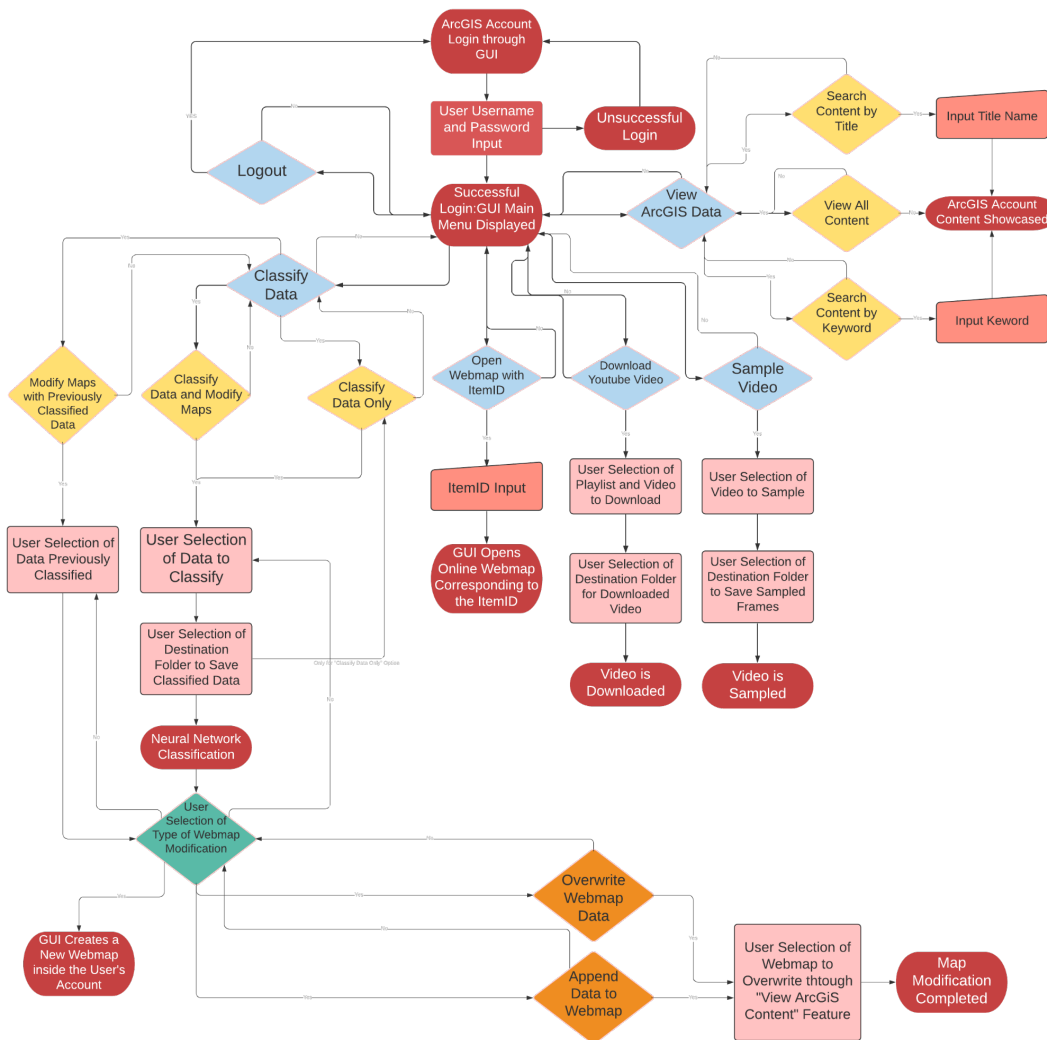


Figure 31. Chart. Complete GUI system diagram

LOGIN AND LOGOUT BUTTON

The first aspect that the GUI implements is the ability to log in and out of the application. The login credentials are the user's online ArcGIS account credential since the ArcGIS API is the backbone of the application, and ArcGIS is where the disaster-damaged road maps are created. Figure 32 shows the main page where the login process starts.

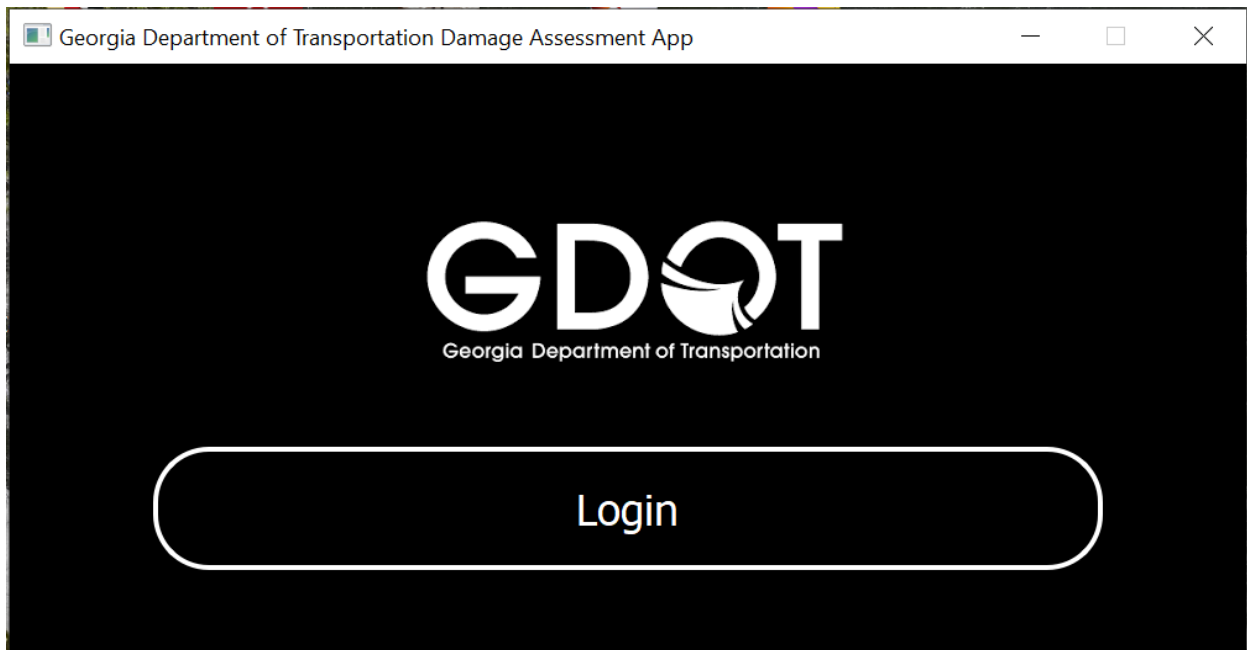


Figure 32. Photo. GUI login window

Once the "Login" button is clicked, the user is asked to enter a username and password, as seen in Figures 33 and 34.

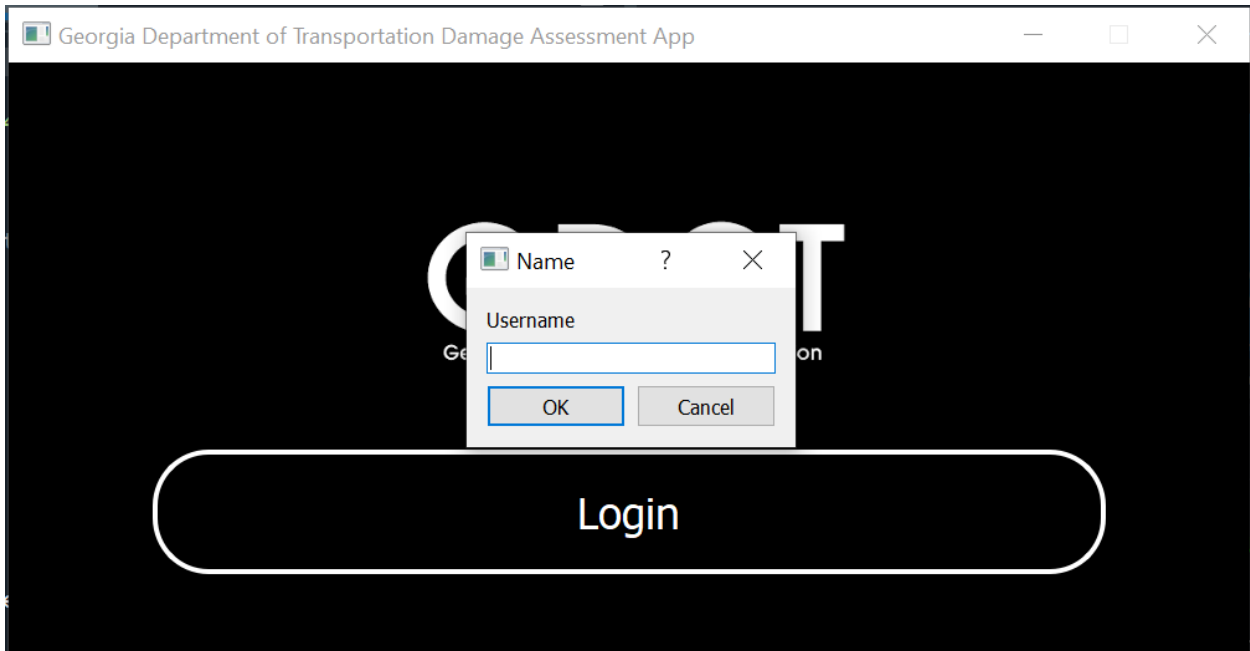


Figure 33. Photo. GUI prompt window for ArcGIS account username

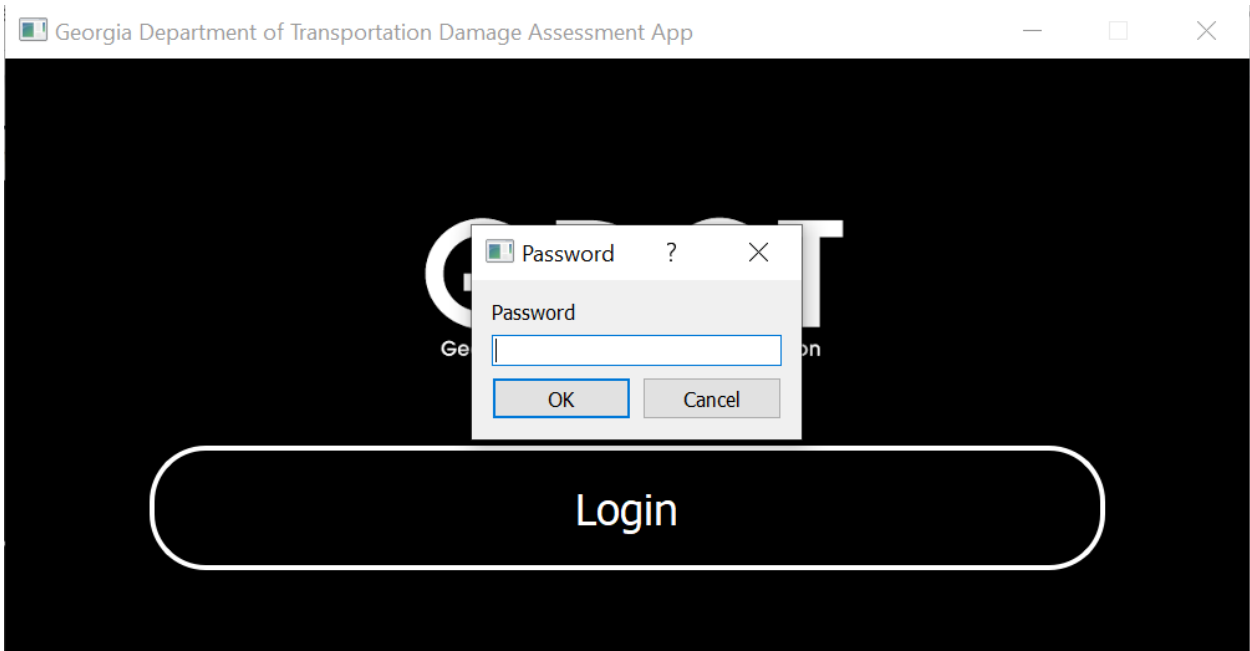


Figure 34. Photo. GUI prompt window for ArcGIS account password

If the user enters an incorrect username or password, the application will indicate that the process was unsuccessful and promote the user to try again, as shown in Figure 35.

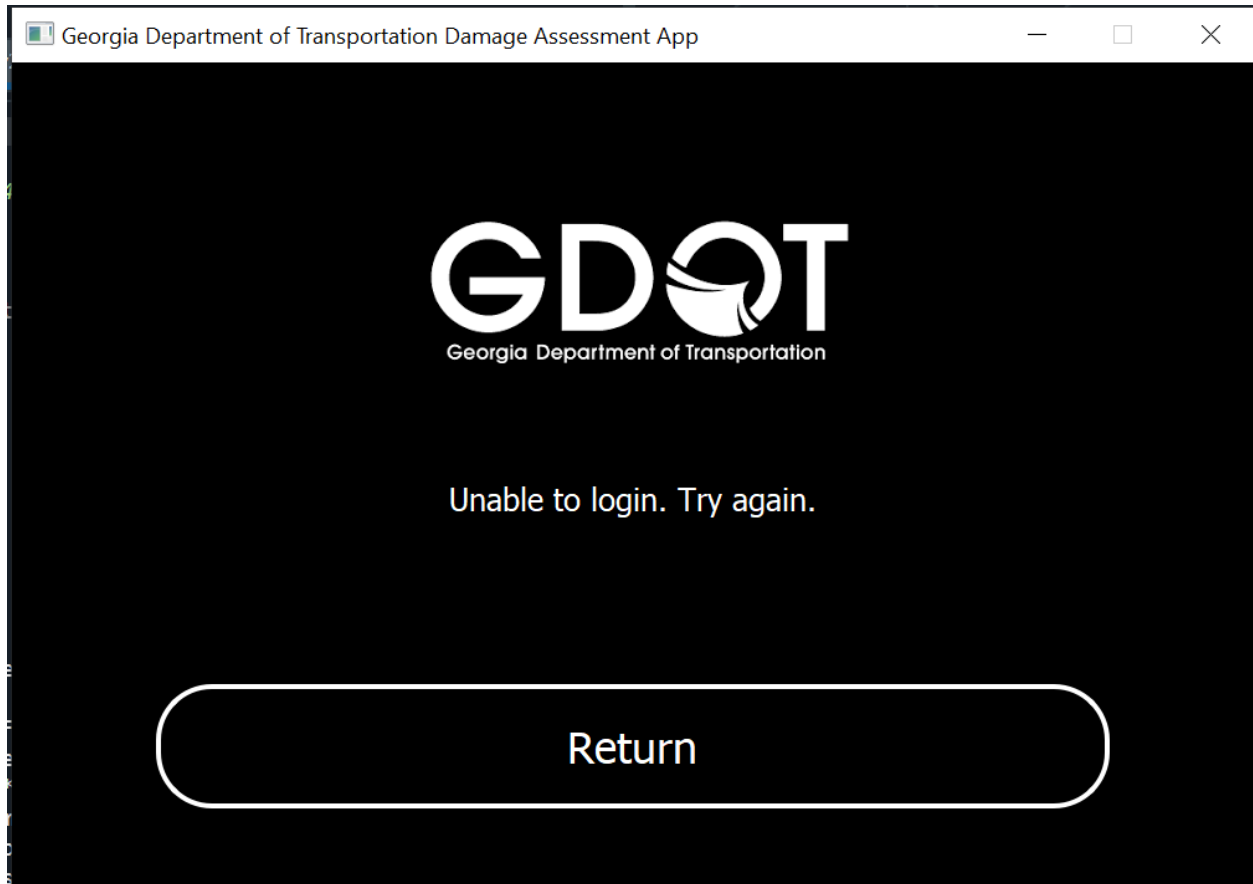


Figure 35. Photo. GUI window for login error

On the other hand, the application's main menu reflects a successful message if the login is successful, as shown in Figure 36. Finally, the user can also log out from the system by clicking on the top left button in the GUI.

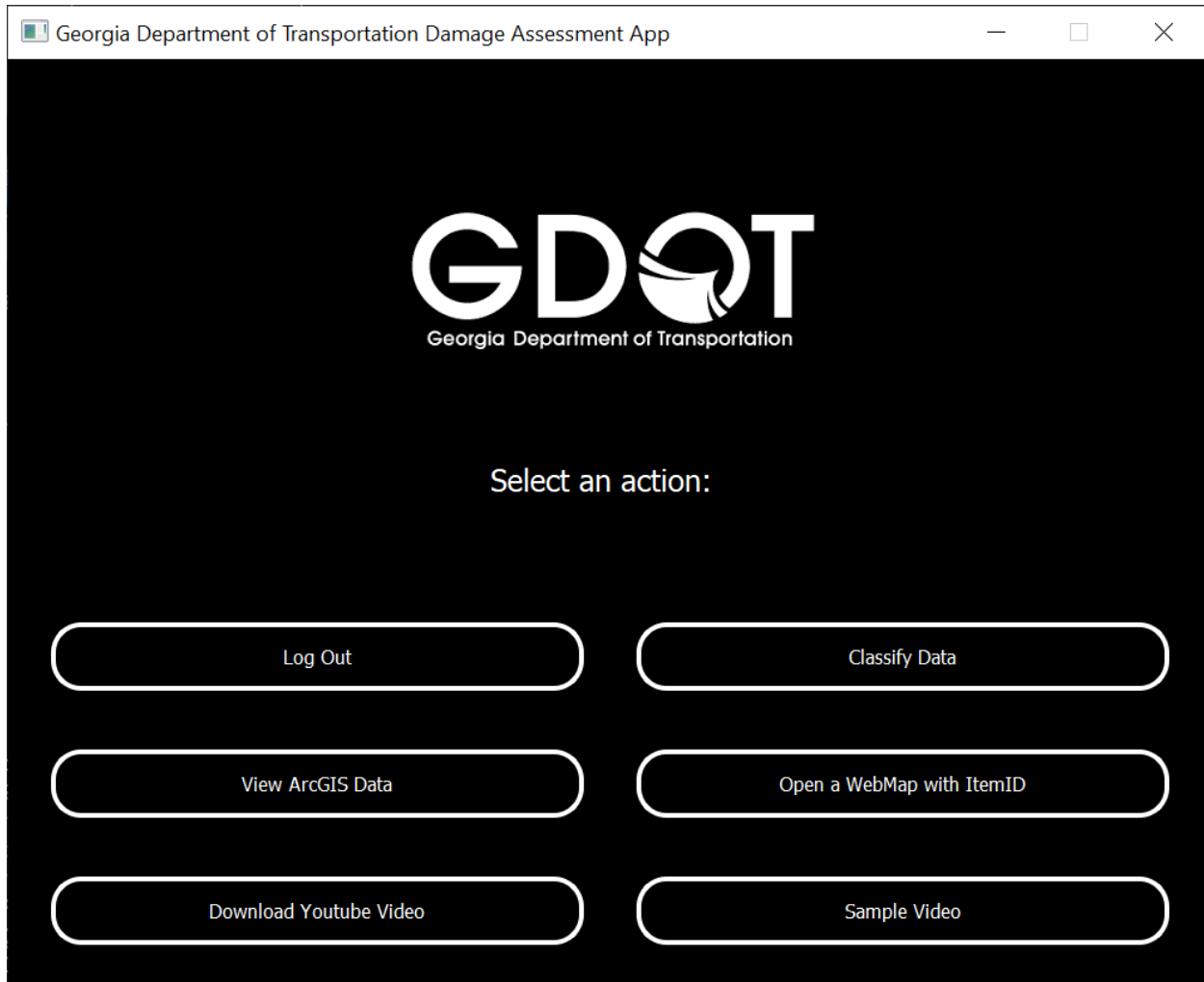


Figure 36. Photo. Application main window

VIEW ARCGIS DATA BUTTON

After successfully logging in, the user is presented with the options illustrated in Figure 37. The user has the option of viewing all of their account data stored in their online ArcGIS account. Once clicked on, this button will take the user to a second window with data viewing options from which the user can pick. This is illustrated in Figure 37.

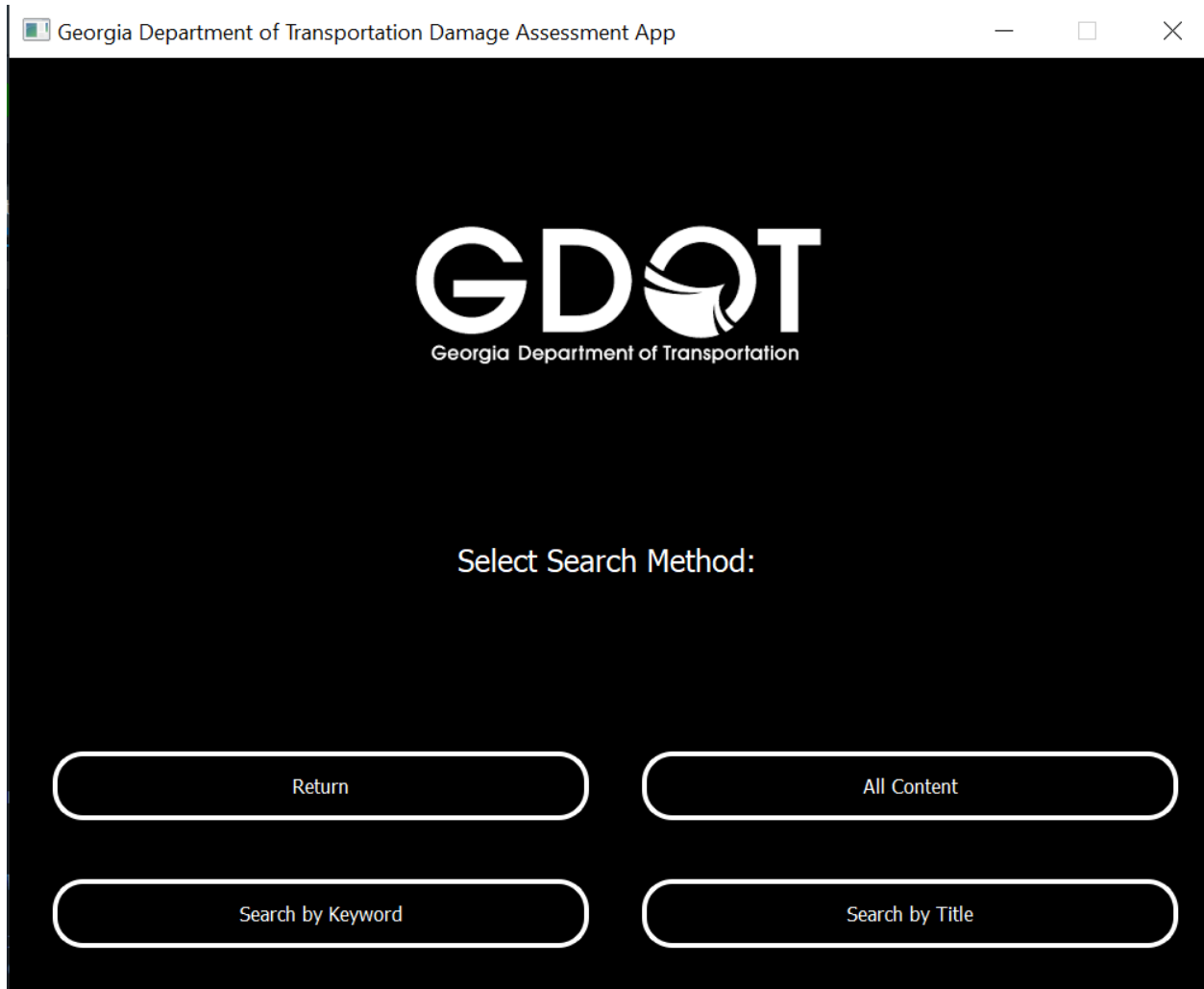


Figure 37. Photo. Data search menu window from view ArcGIS data

In the second window, the user is introduced to four options, which are “Return”, “All Content”, “Search by Keyword”, and “Search by Title”. The “Return” button allows the user to return to the main menu window. The “All Content” Button enables users to see all their existing content in their ArcGIS account, as illustrated in Figure 38.

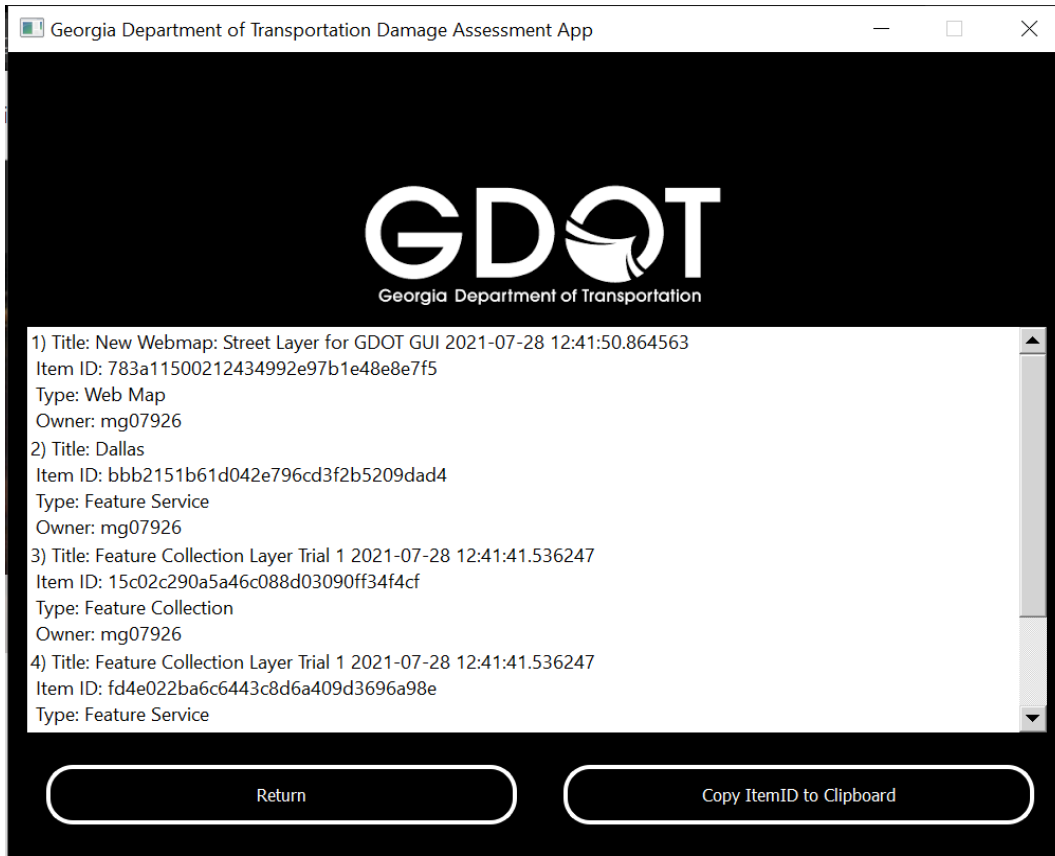


Figure 38. Photo. All ArcGIS account content window

Additionally, this window allows the user to grab the selected service’s Item ID with the “*Copy ItemID to Clipboard*” option. This feature will be necessary when the user wants to open webmaps directly from the GUI. It is worth noting that the GUI will only open webmaps and not other features such as Feature Layers.

Once back to the previous data search window in Figure 37, the user can also search content in their account through keywords or item titles. These implementations are accessed through the “*Search by Keyword*” and “*Search by Title*” buttons. Figures 39 and 41 showcase the prompt window for searching the data using a keyword and title, respectively, while Figures 40 and 42 show the search results for both search techniques.

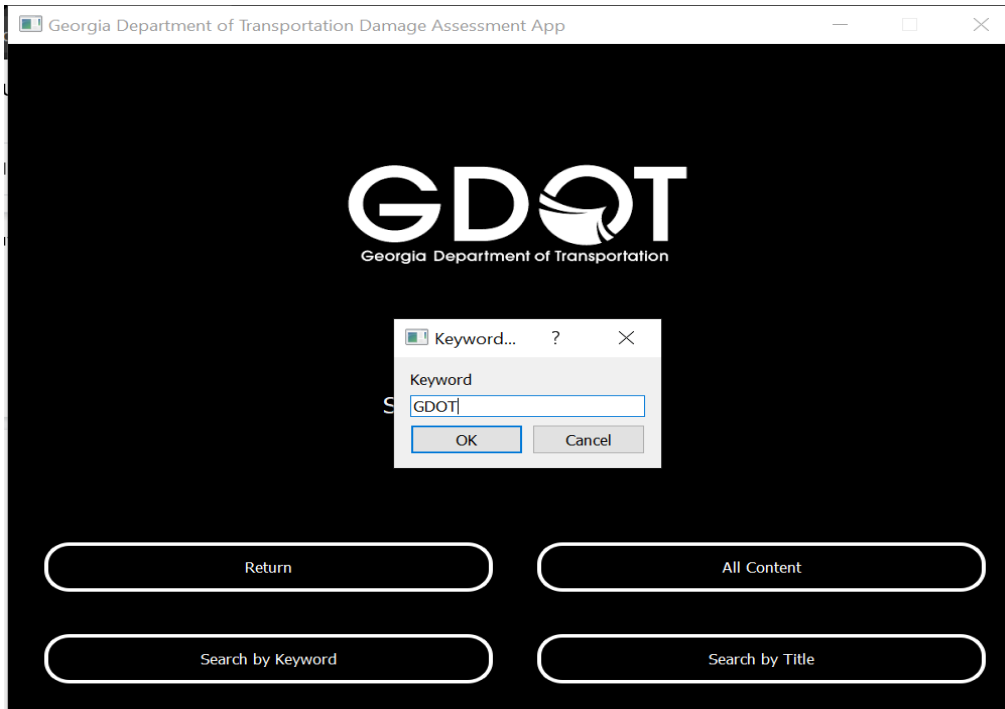


Figure 39. Photo. Keyword search prompt window for ArcGIS account content

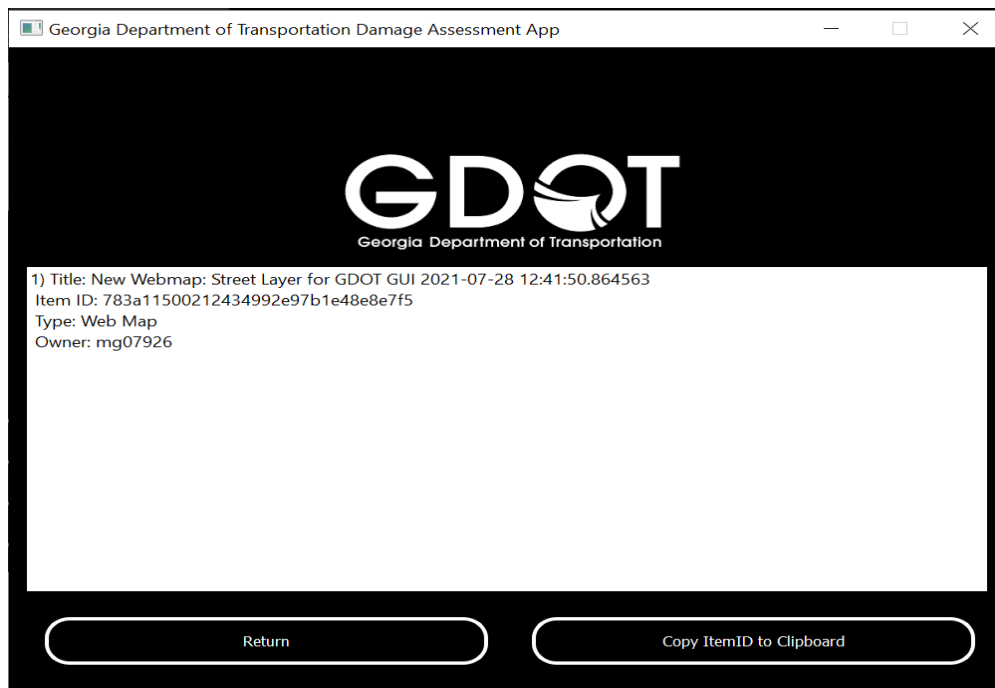


Figure 40. Photo. Search results window for keyword content search

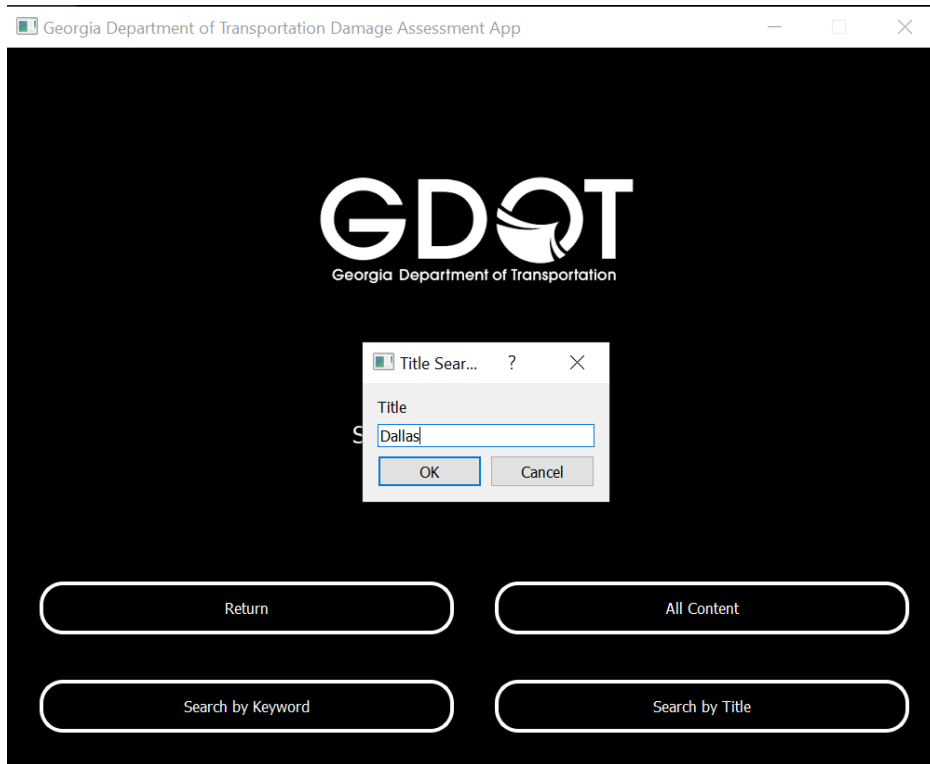


Figure 41. Photo. Title search prompt window for ArcGIS account content

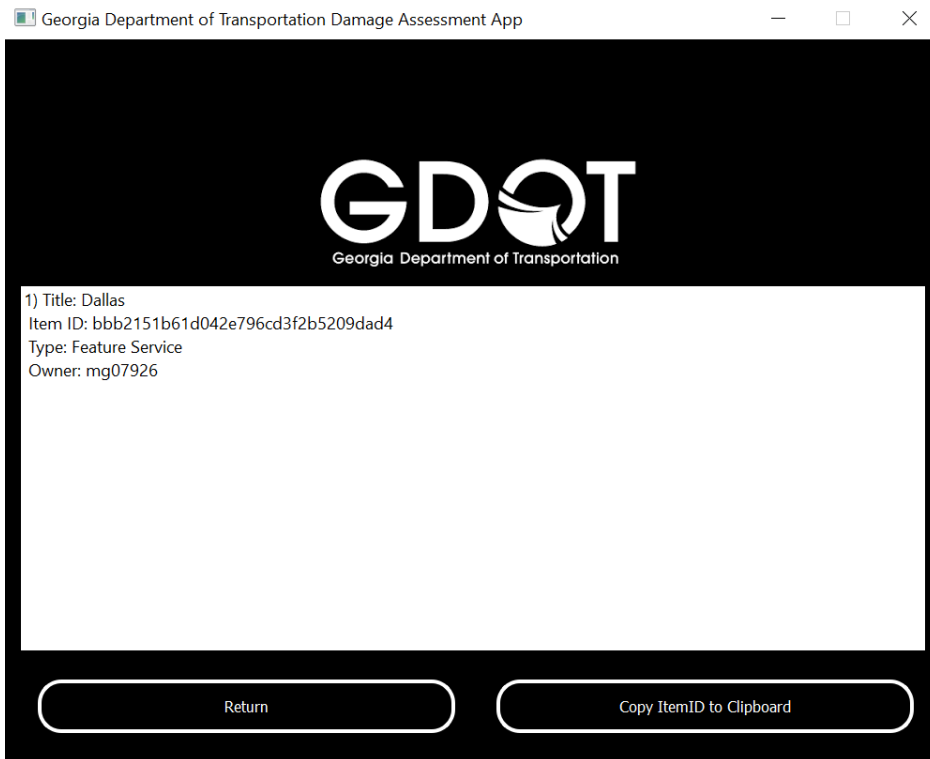


Figure 42. Photo. Search results window for title content search

Similarly, both these options show windows that have the option to copy an item's ID to the user's clipboard. The options presented in this section allow the user to move their data around in ArcGIS without switching environments.

OPEN WEBMAP WITH ITEMID BUTTON

The next feature implemented in the GUI is the ability to open webmaps directly from the GUI. The user can input a webmap's item ID by clicking on this button, so the GUI opens the set web browser to showcase a webmap with layers. If the webmap's item ID for a specific map is not known, it can be obtained by copying it to the clipboard by searching the ArcGIS data through the process explained in the previous section. This process of retrieving the webmap item ID and accessing it is illustrated in Figures 43, 44, and 45.

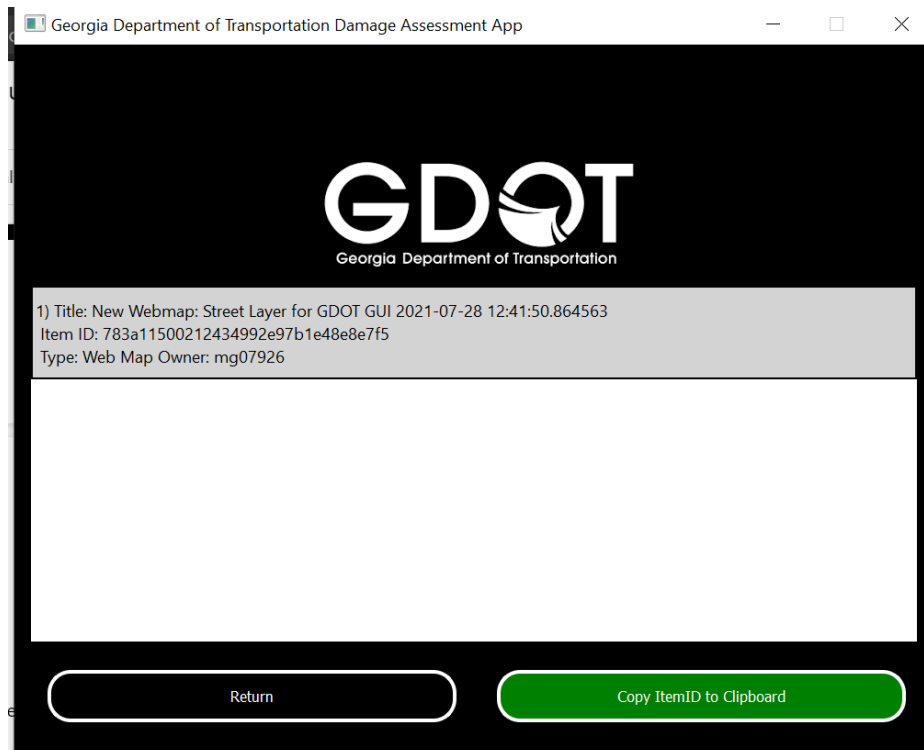


Figure 43. Photo. Copy ItemID to clipboard from ArcGIS account content window

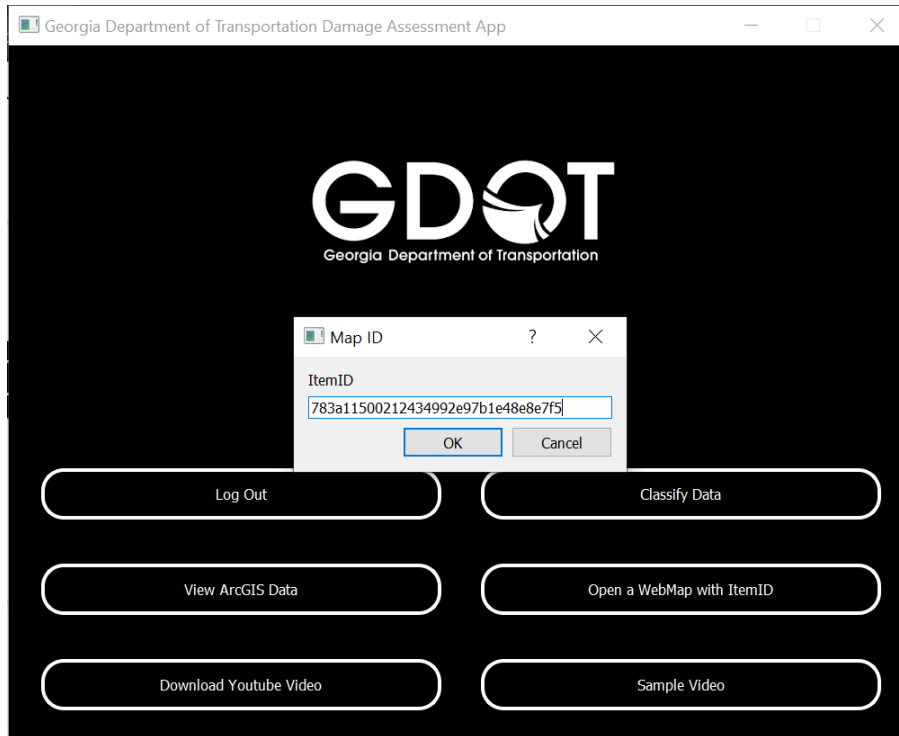


Figure 44. Photo. Paste ItemID from clipboard to map ID prompt window

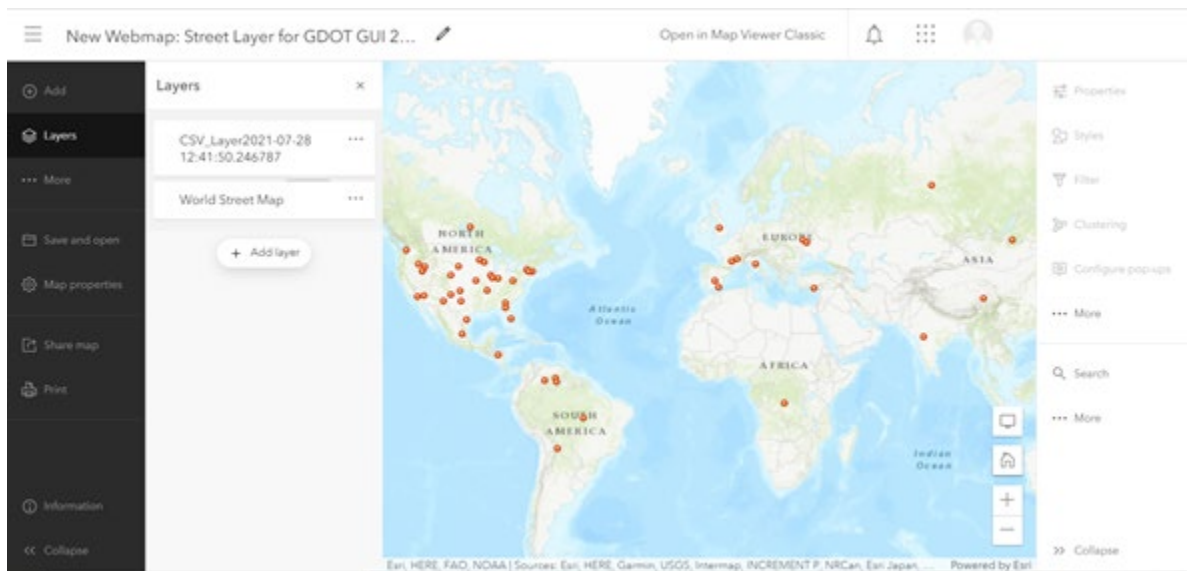


Figure 45. Map. ArcGIS webmap opened using the “Open Webmap with ItemID” button

This feature is particularly useful for anyone who only wants to visualize the maps/layers. The user can open any webmap and obtain any webmap's item ID directly through navigating the application.

CLASSIFY DATA BUTTON

Once back to the main application window in Figure 36, the essential feature of the application is classifying the data obtained from the UAV missions. When the “*Classify Data*” button is clicked, a second window is opened where there are three options from which the user can choose to classify the data and map the classification once it is completed. Figure 46 shows the second window with the classification and data mapping options.

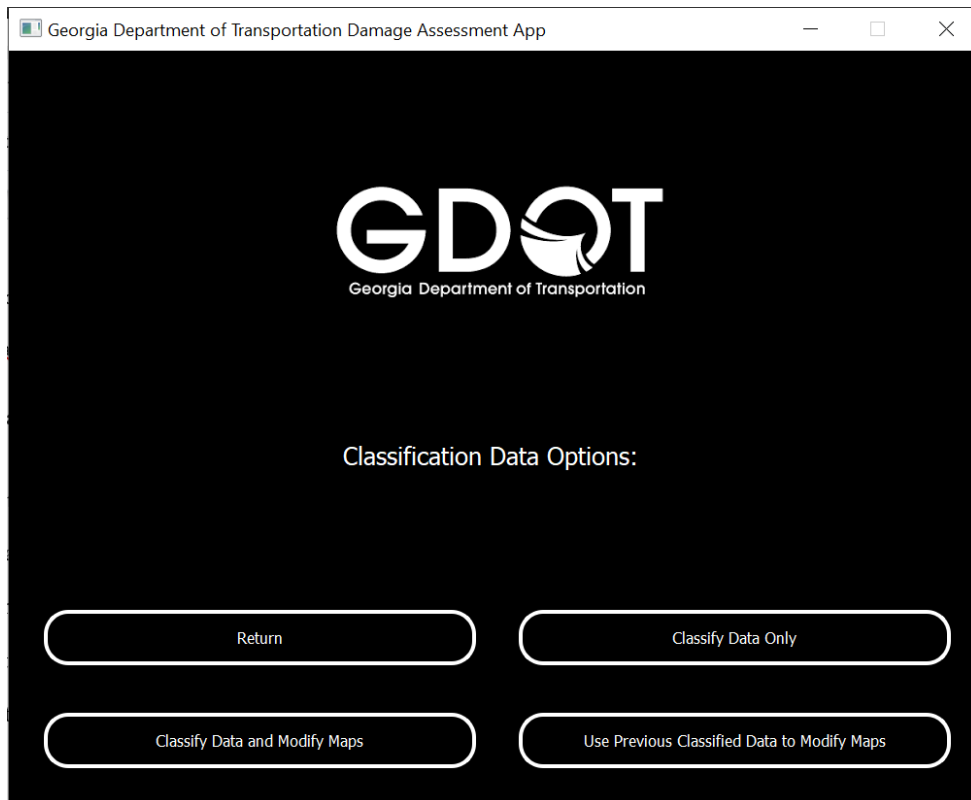


Figure 46. Photo. Data classification and mapping window

Classify Data Only Button

The “*Classify Data Only*” button enables the user to classify images from a Google Drive folder or any other file system the ground control and processing server uses, as seen in Figure 47. The input folder selected should have the sampled images of the unmanned aerial vehicle’s mission video stream. After selecting the input folder, the user is prompted to select where the classification results should be saved, as seen in Figure 48. This process can be automated through a task scheduler.

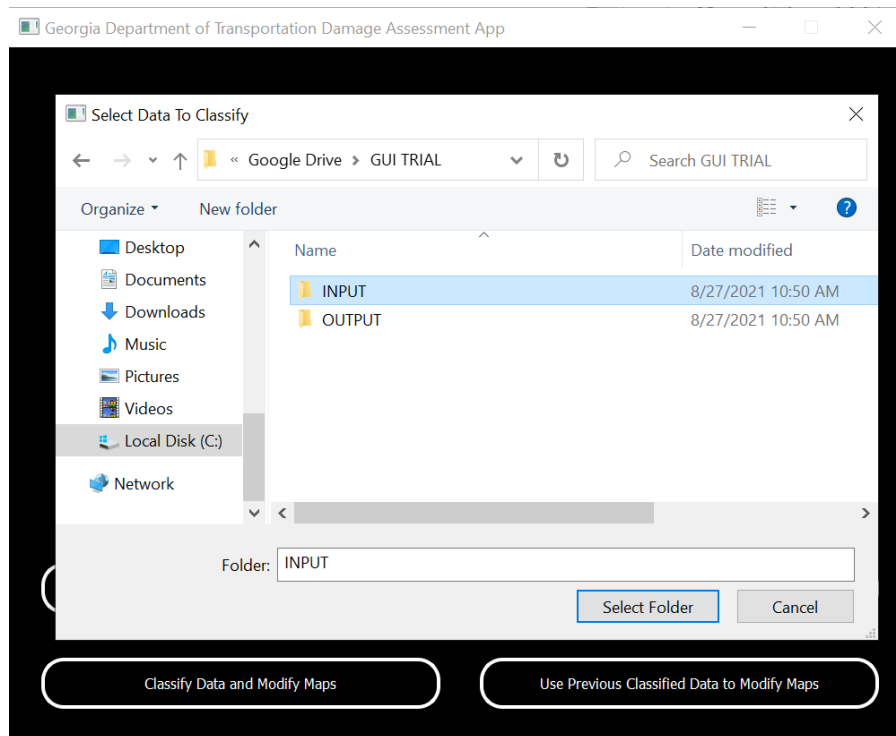


Figure 47. Photo. Data selection for classification

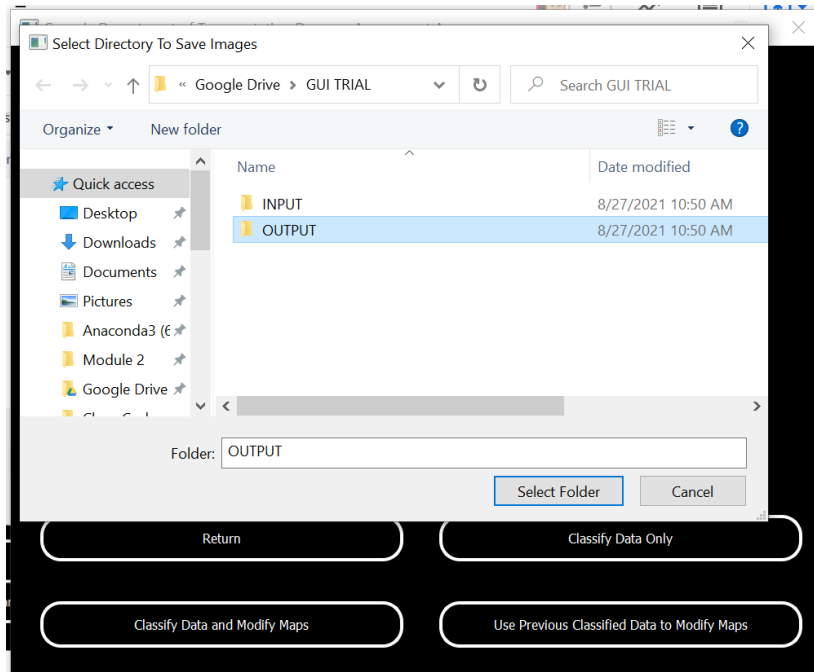


Figure 48. Photo. Folder selection to save classification output and results

The classification results will include CSV and XLSX files summarizing all the classification results with their associated geo-tagging information, as well as a zip file with the disaster-damaged classified images, as seen in Figure 49.

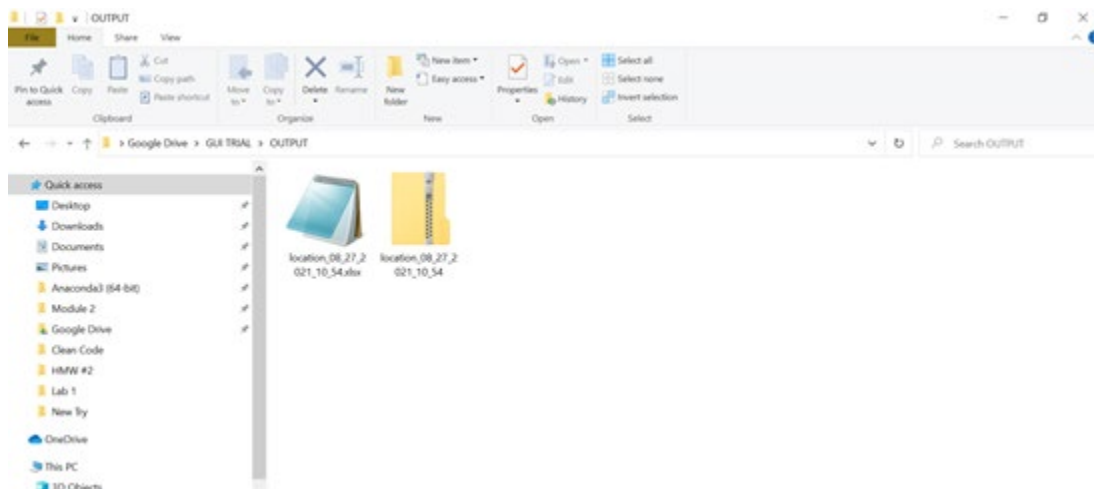


Figure 49. Photo. Saved data and results in user selected location

Moreover, the application will notify the user whether the classification was successful without the need to access the files to inquire about that information, as seen in Figures 50 and 51.

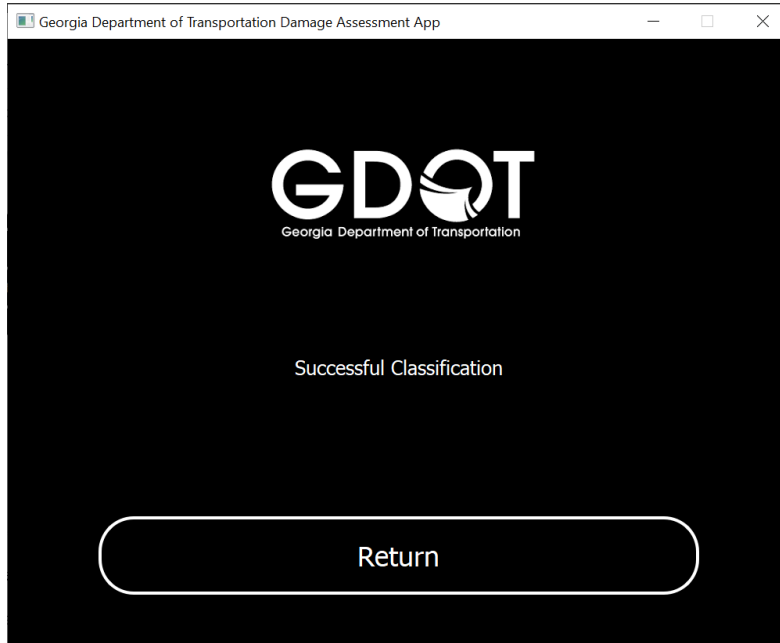


Figure 50. Photo. Successful classification window

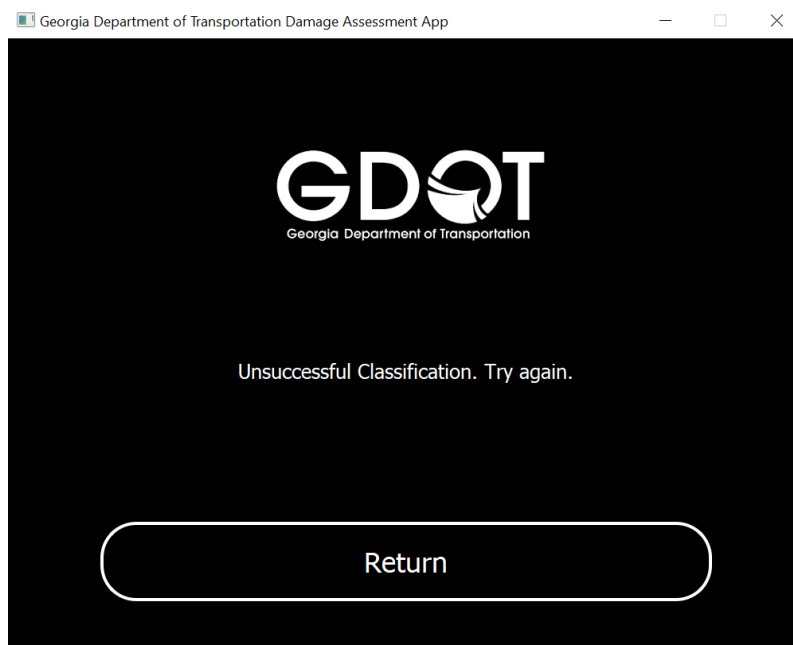


Figure 51. Photo. Unsuccessful classification window

Classify Data and Modify Maps Button

On the other hand, the “*Classify Data and Modify Maps*” button enables the user to perform the same data classification process as described in the previous section. However, it also provides the option of creating a new webmap, overwriting an existing webmap, or appending a previously created webmap with the information obtained from the classification process. These options are illustrated in Figure 52.

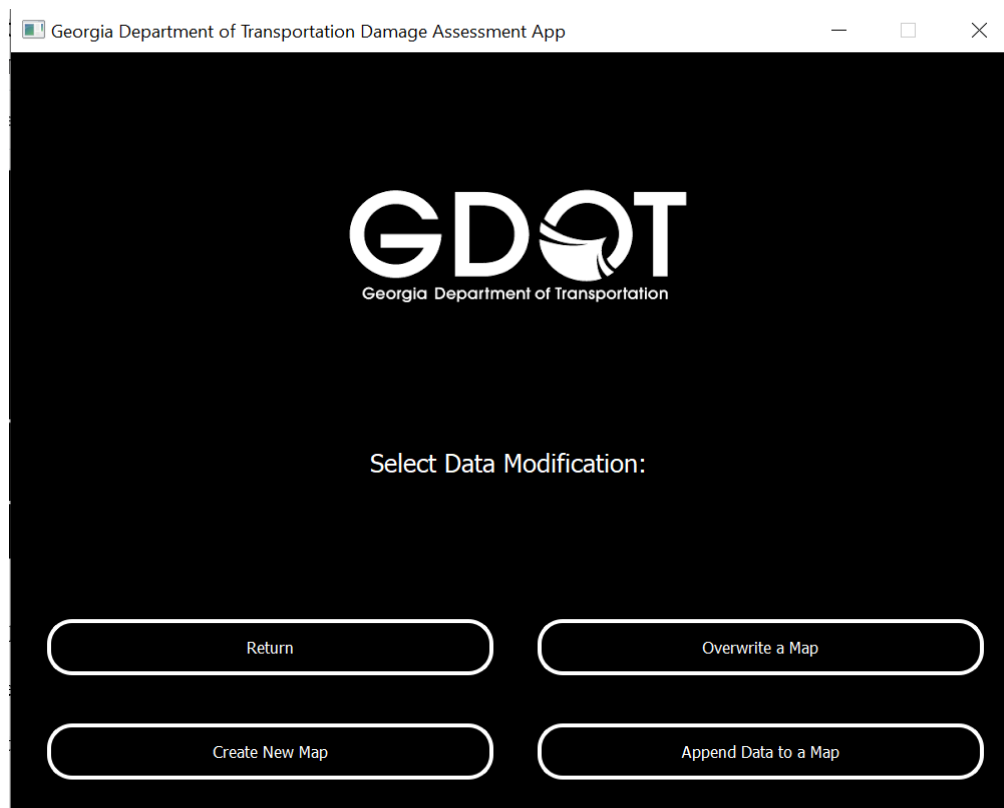


Figure 52. Photo. Webmap modification menu window after data classification

The “*Create New Map*” button simply creates a new webmap for the user in their ArcGIS account. The other two options, “*Overwrite a Map*” and “*Append Data to a Map*”, prompt the user to select an existing ArcGIS webmap. The GUI enables the user to browse through all the existing ArcGIS webmaps and select the webmap that needs to be modified. This process is illustrated in Figures 53 and 54. In those figures, the GUI windows shown, as a result of choosing to overwrite a map

or append information to a map, resemble those of the “View ArcGIS Data” button from the main window since both options require accessing data in the ArcGIS account.

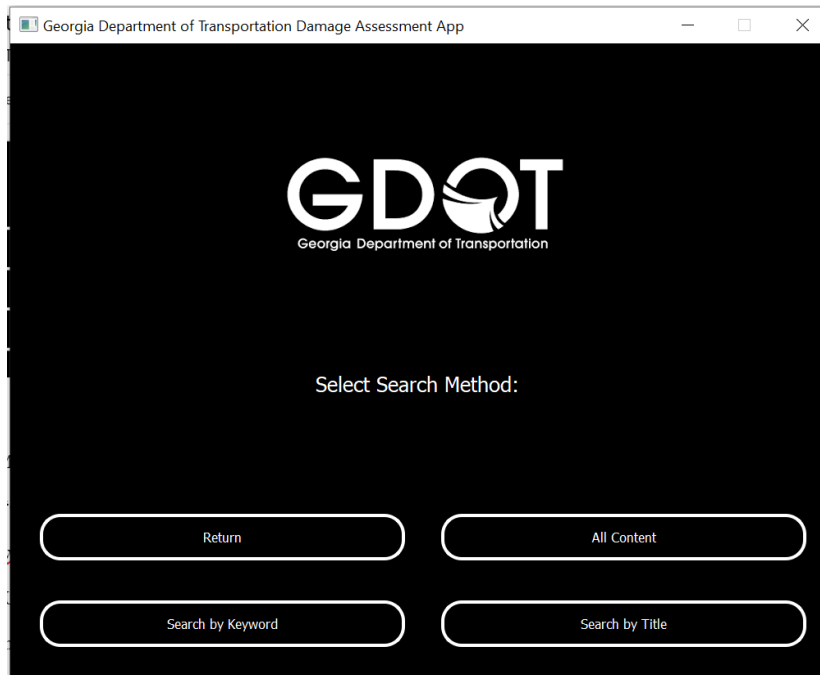


Figure 53. Photo. Webmap search window after data classification and selection of webmap modification

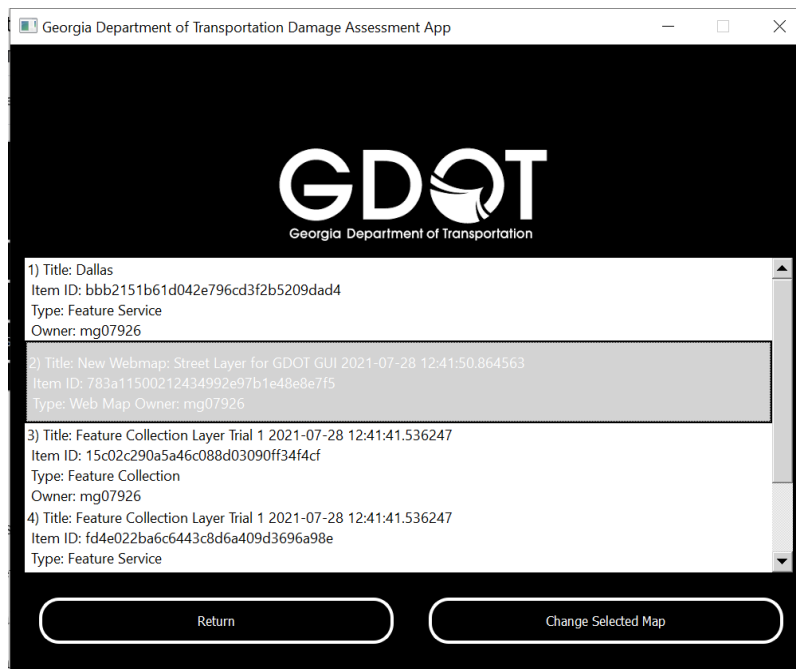


Figure 54. Photo. Webmap selection for modification of data

After selecting the desired webmap to be modified, the “*Change Selected Map*” button is clicked to initiate the modification. This will prompt the user to select the classification output CSV file obtained from the classification process, as seen in Figure 55.

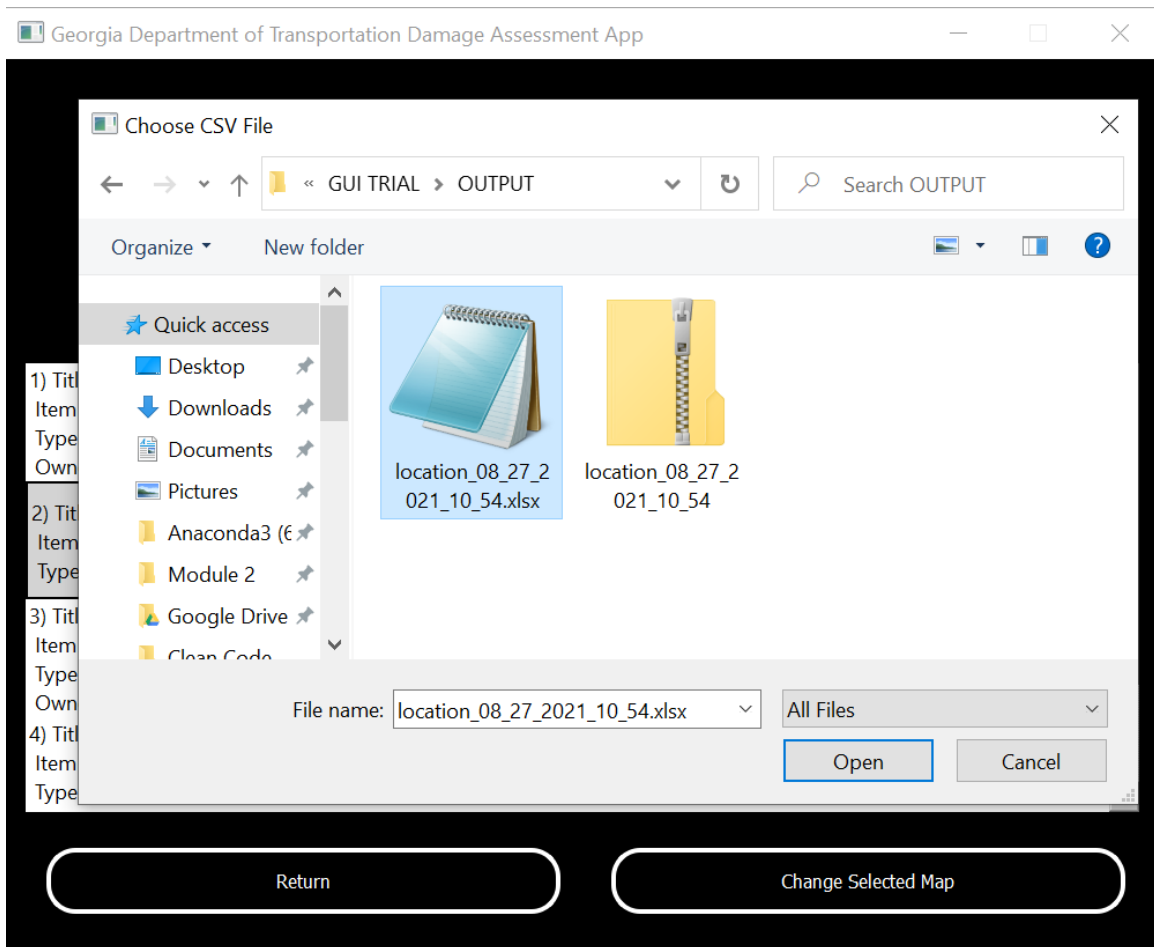


Figure 55. Photo. User selection of data to modify webmaps

When the modification process completes, the main window will display an appropriate message for the user, as illustrated in Figure 56. It is also worth noting that the “*Overwrite a Map*” button completely overwrites the webmap’s layers and adds “*Overwritten*” to the webmap’s title. In contrast, the “*Append Data to a Map*” button adds more information to the webmap’s data layers and adds “*New Data Appended*” to the webmap’s title. The resulting maps can be viewed using the “*View ArcGIS Data*” button, as depicted in Figure 57.

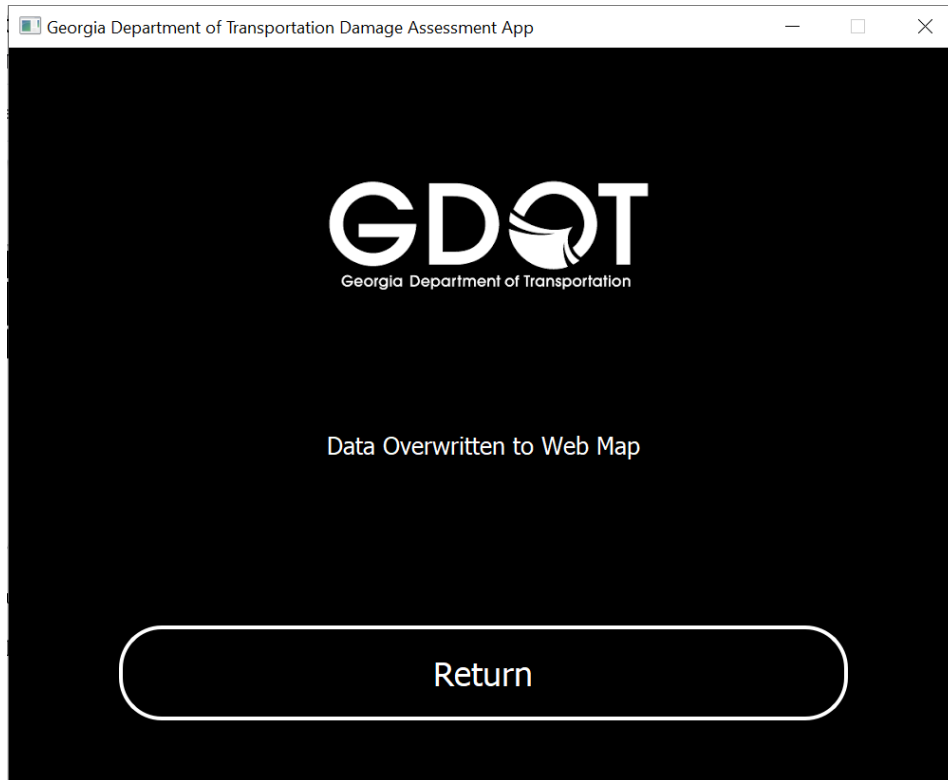


Figure 56. Photo. Data successfully overwritten window

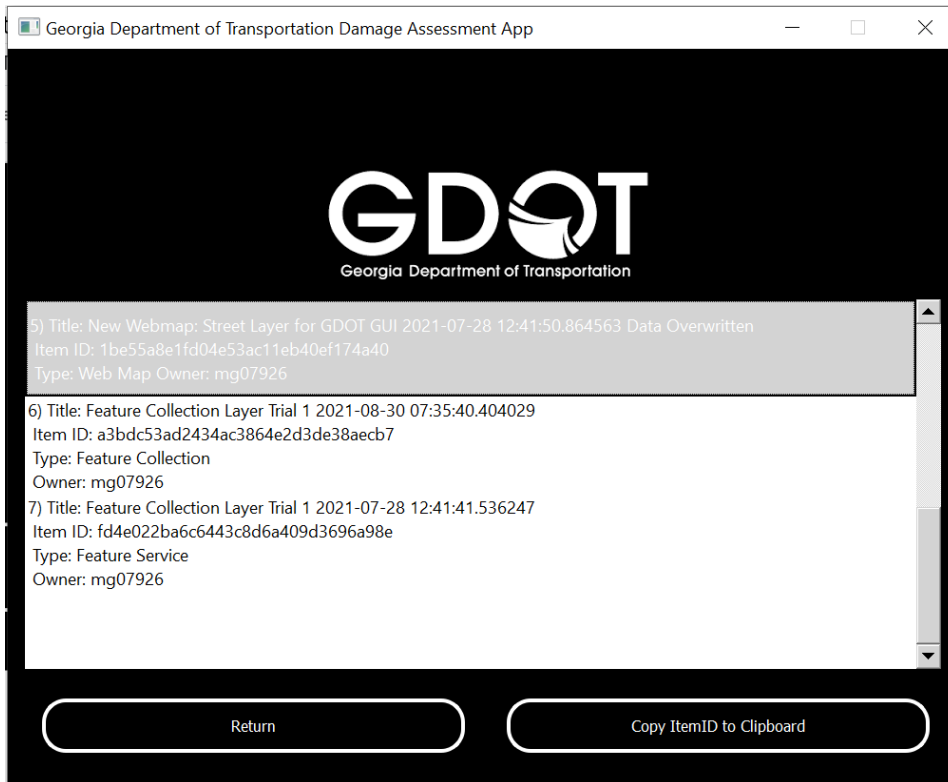


Figure 57. Photo. Overwritten map displayed in the content of the user's GUI account

Use Previously Classified Data and Modify Maps Button

The “*Use Previous Classified Data and Modify Maps*” button enables the user to use previously classified data to modify webmaps in ArcGIS. This feature implements the same options as the “*Classify Data and Modify Maps Button*” regarding what type of modification the user can do, as seen in Figure 52 before. The “*Use Previous Classified Data and Modify Maps*” and “*Classify Data and Modify Maps*” buttons’ follow the same process; the only distinction is the input source: the former uses already classified data to modify a map, while the latter classifies the data and uses these results to modify a map.

DOWNLOAD YOUTUBE VIDEO BUTTON

The main application window, illustrated in Figure 36, allows downloading videos initially streamed into the YouTube platform with the RTMP communication protocol. This feature was added to demonstrate that RTMP streamed video feed can be accessed, classified, and added to an ArcGIS map. YouTube is used as an RTMP existing server to demonstrate this feature, which can be easily reconfigured to connect to any other RTMP server. This feature checks the existing playlists of a predetermined YouTube account set up internally. This process is illustrated in Figure 58.

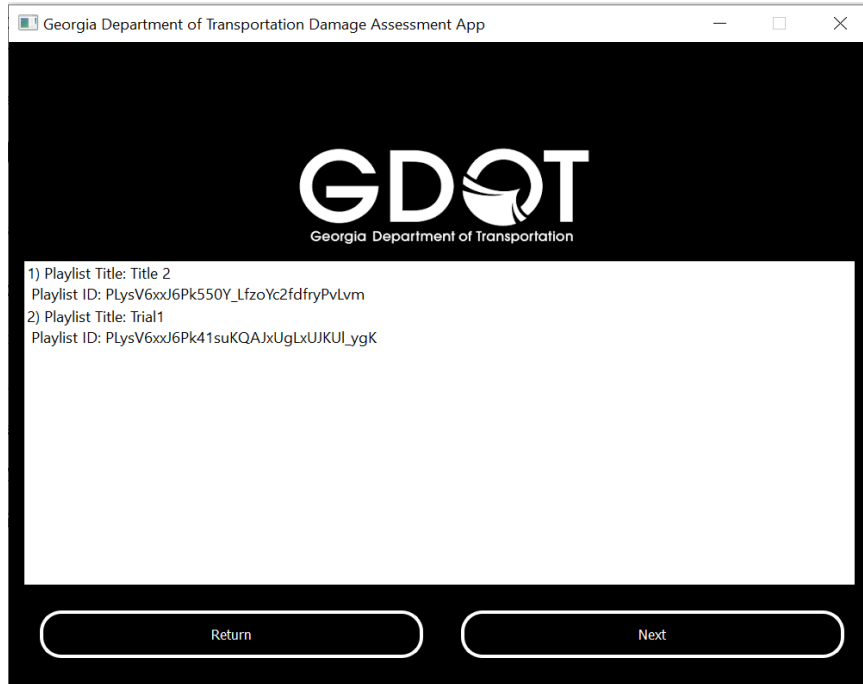


Figure 58. Photo. Selection of available YouTube playlists

Once the desired playlist is selected, a second window will appear. In this window, all the videos corresponding to that playlist are listed. To download a video, it first has to be selected, and then the user simply clicks on the “*Download Video*” button, as illustrated in Figure 59.

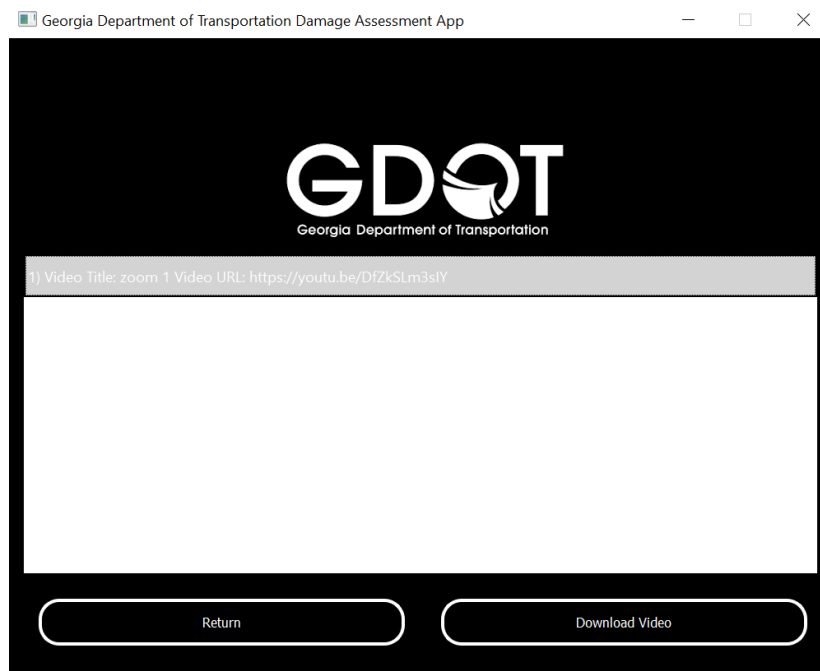


Figure 59. Photo. Selection of available videos found in the YouTube playlist

After clicking the “*Download Video*” button, the GUI will prompt the user to select where the video will be saved. This feature allows for the navigation of the file system to select where this video will be saved. This process is illustrated in Figure 60.

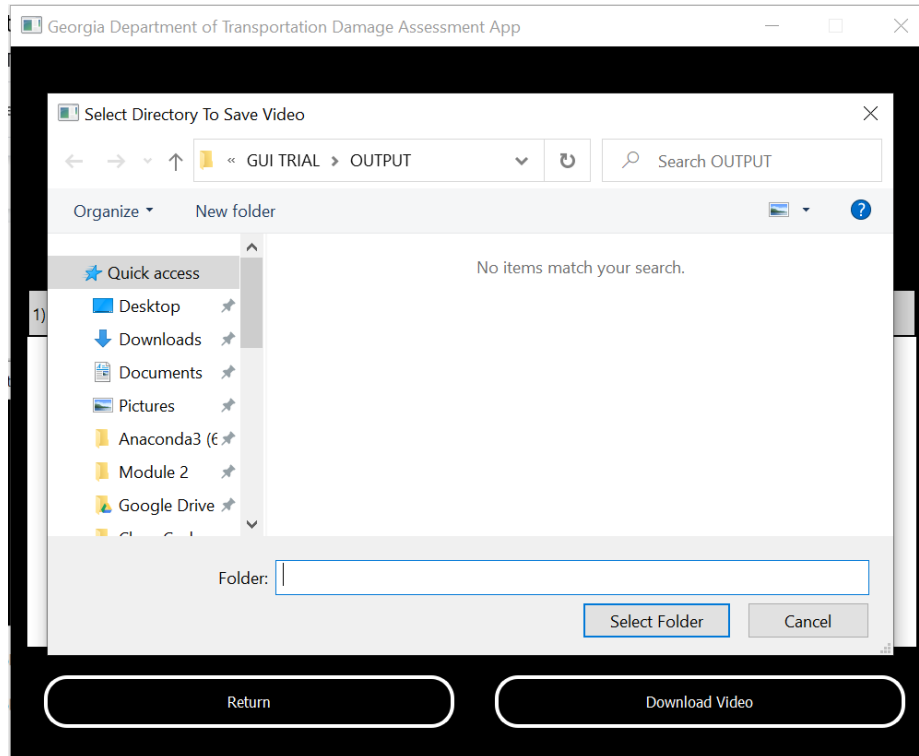


Figure 60. Photo. Directory selection to save a downloaded video

Once the destination folder where the video will be saved is selected, the application will download the video and open up another window indicating whether the download has successfully been completed or not, as depicted in Figure 61. Finally, if the video was downloaded successfully, it will appear in the desired directory, and the user will be able to access it when necessary, as shown in Figure 62.

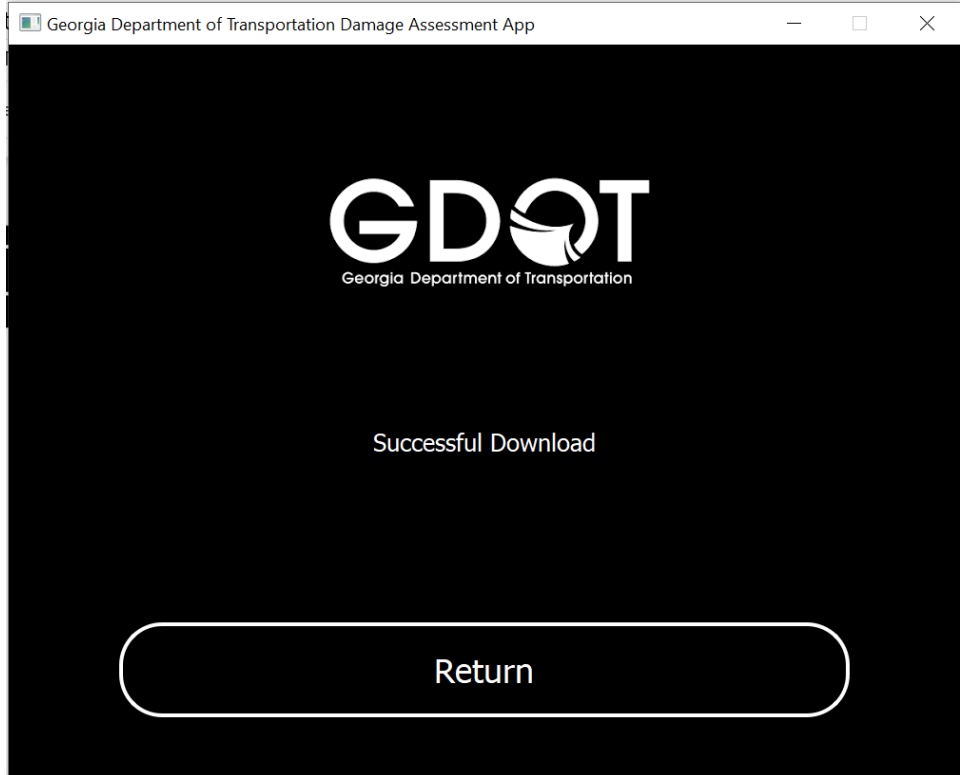


Figure 61. Photo. Video successfully downloaded window

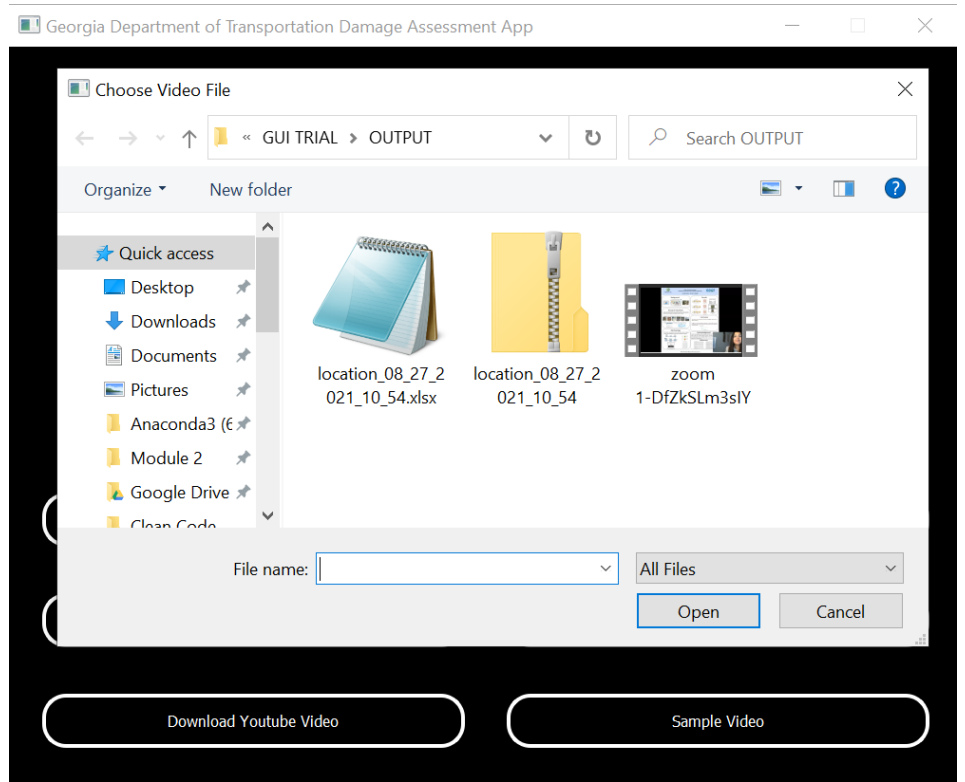


Figure 62. Photo. Downloaded YouTube video

Sample Video Button

The “*Sample Video*” button enables the user to sample any video provided by the UAV (or any other video source). When used, this feature prompts the user to select the video that needs to be sampled and the destination directory where these samples will be saved, as illustrated in Figures 62 and 63.

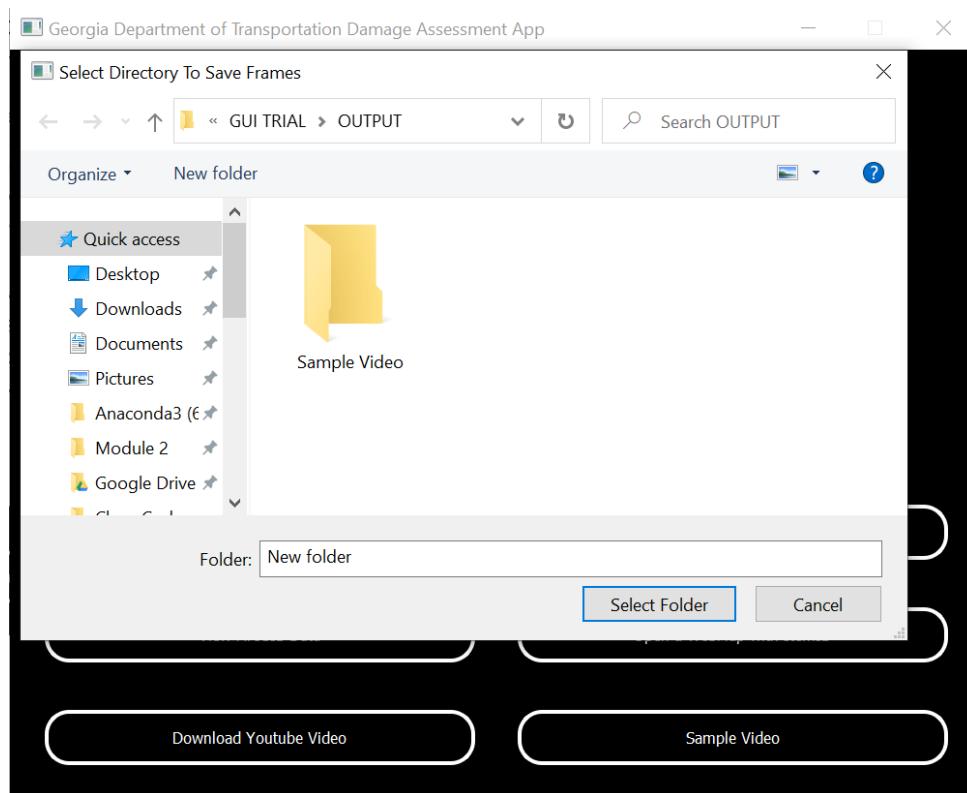


Figure 63. Photo. Directory selection to save sampled frames from video

Currently, the sampling period is set to 40 frames; however, it can be reconfigured if needed. When the video is selected, the application will start sampling the video and saving the frames to the previously selected directory, as depicted in Figure 64.

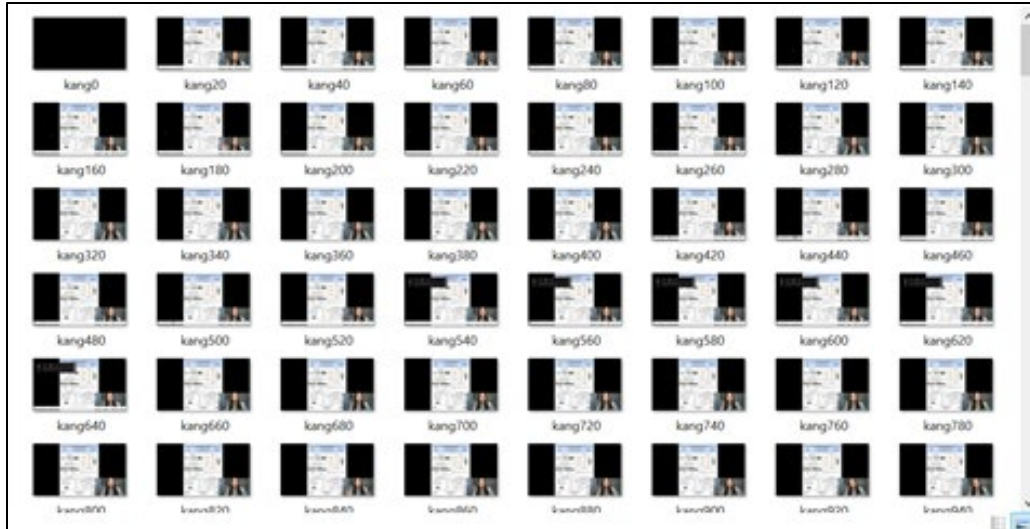


Figure 64. Photo. Output directory with the sampled video frames

Once the sampling process is completed, the application will open a window indicating the completion of the sampling process, as depicted in Figure 65. For automation purposes, the process of downloading a video from an RTMP server, saving it locally, and sampling it can be automated through a task scheduler.

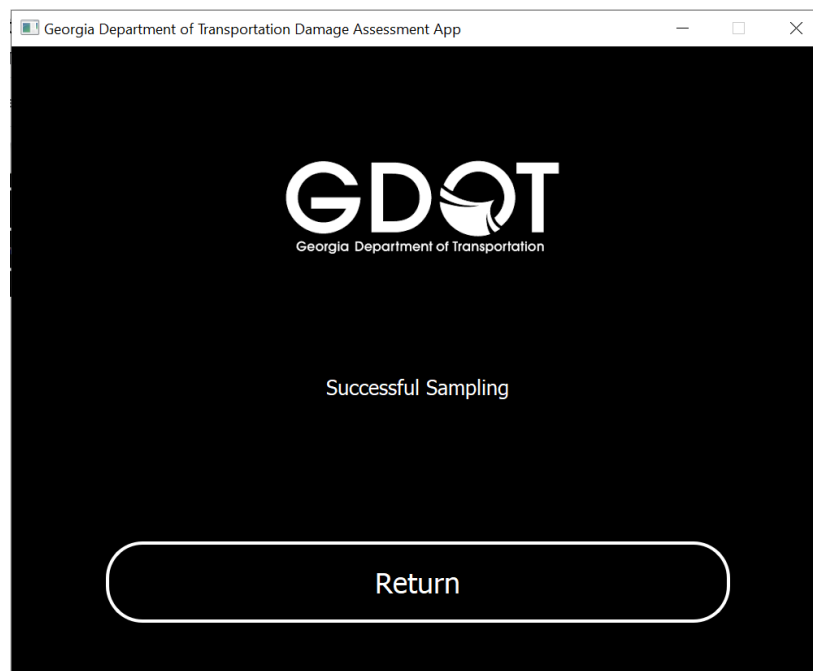


Figure 65. Photo. Video successfully sampled window

CHAPTER 5. CLASSIFICATION RESULTS AND DISCUSSION

EXPERIMENT DESCRIPTION

For testing the effectiveness of classifying disaster-damaged roads, a total of three different convolutional neural networks (CNNs), AlexNet, GoogLeNet, and ResNet50, were investigated. Parameters, such as mini-batch size, max epochs, and learning rate, were set for each network. The mini-batch size defines the number of samples run in an epoch. The number of epochs is a hyperparameter that sets how many complete trials the network must complete during training. The learning rate in a neural network controls the amount of change in the model whenever the weights are updated, determining how quickly or slowly a neural network model learns.

All networks were tested with a mini-batch size of 64, 15 epochs, and a learning rate of 10^{-4} . Additionally, all networks required a specific image size to work properly; in all cases, the input was $224 \times 224 \times 3$. To accommodate this requirement, a MATLAB function was created to resize all input images. Once the resizing was complete and the parameters tuned, transfer learning was applied to the networks. Transfer learning is a machine learning process of selectively changing the output categories of a pre-trained classification network to repurpose and customize the classification to classify a different set of tasks. For example, a network trained on a large dataset to classify different types of plants can be repurposed to classify a smaller dataset of animals. To achieve this result, the network's knowledge obtained by being pre-trained with a large dataset is kept unchanged in all the network layers except the final few layers. The last few layers of the network are retrained to avoid overfitting and obtain more specific features from the smaller dataset. Transfer learning addresses the time constraints of building large data sets and performing supervised learning by redirecting pre-trained networks' classification layers. Furthermore, transfer learning tackles the hardware costs, e.g., GPU, necessary to perform high computational

analysis with large datasets, considering that only the last layers need retraining when repurposing a network.⁽⁹⁾ These unique advantages are integral to the functionality and competitiveness of classification systems, lowering hardware requirements while improving capabilities.

The networks were tested and validated using the dataset from the image library compiled for this project once the transfer learning process was completed. In this step, 4-fold cross-validation was used by partitioning the dataset into four subsets. Cross-validation is a machine learning validation technique used to evaluate the performance of the CNN by averaging the performance of the network using different subsets of the dataset, in a very systematic way, to obtain true network performance. This method is also useful for avoiding overfitting issues due to training and testing on a limited dataset of disaster-damaged roads. When performing k -fold cross-validation on a dataset, the method randomly divides the data into k portions. In the case of this work, the dataset was divided into four folds. Then, three folds are selected to be part of the training and validating dataset, and the remaining fold is left to be used as the testing dataset. This procedure is repeated four times to allow each portion to be used as a test dataset. Therefore, the network is trained and analyzed four times. The analyzed results (F1 score, Precision, Recall) of all the training and testing datasets are then averaged by obtaining their mean. This method allows for a less biased model.

CLASSIFICATION QUALITY MEASURES

The primary measure of quality when considering classification performance is the accuracy of the classified categories. Numerous image classification assessment measures exist to objectively calculate the classification performance. Due to the nature of the work and the limitations of a small dataset, the original images are run through cross-validation and transfer learning processes

to avoid overfitting. The classification results are then analyzed with four quantitative methods. This particular work utilizes precision, recall, specificity, and F1 score analysis methods to ensure the requirements of proper classification are met.⁽⁸⁾

Classification Outcomes

To be able to define these measures, the classification outcomes need to be defined first. There are four possible category classification accuracy outcomes: true positive (TP), false positive (FP), true negative (TN), and false-negative (FN). A true positive outcome results from the CNN's classification output of an image matching that image's actual category. Furthermore, a true negative is the CNN correctly classifying an image as not part of another category other than its own. On the other hand, a false positive is the CNN classifying an image as belonging to a category that is not the image's actual category. Finally, a false negative means incorrectly classifying the image as not belonging to its actual category.

Quality Measure – Recall

Recall, also called sensitivity, refers to the CNN's ability to correctly classify images of a category as part of that category. This method is designed to compare the capability of a network to discern images belonging to a specific category properly. It functions by calculating the number of true positives classified cases divided by all the true positives and false negatives classified, as indicated in eq. 1.

$$Recall = \frac{TP}{TP+FN} \quad (1)$$

Quality Measure – Precision

Precision refers to the network's classification accuracy of positive cases for each existing category, as shown in eq.2. This measure demonstrates the percentage of true positive classifications out of all the positive classifications obtained by the network in each category.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Quality Measure – F1 Score

The F1 score is the harmonic mean between statistical precision and statistical recall, as shown in eq.3. This measurement is better suited for measuring incorrectly classified cases by a network and is represented by a number between 0 and 1. The F1 score also proves helpful when there is an imbalanced category distribution in the dataset.

$$F1 = \frac{2(Precision)(Recall)}{Precision+Recall} = \frac{2TP}{2TP+FP+FN} \quad (3)$$

Quality Measure – Specificity

Finally, specificity refers to the percentage of images correctly classified as not part of a category. This allows us to assess how the network performs at deciding if an image is truly part of a category or not. The formula to find specificity is demonstrated in eq.4.

$$Specificity = \frac{TN}{TN+FP} \quad (4)$$

SIMULATION RESULTS AND DISCUSSION

The results presented in this section were produced from running computer simulations of the classification neural networks. The referenced classification measures of F1 score, precision, specificity, and recall were chosen due to their capabilities in assessing vital quality aspects of the image classification achieved by the networks tested. Though each measure does not show the complete picture, presented together, they allow for a better understanding of the results. Moreover, these methods are four of the most commonly used and referenced classification measures, allowing replicability and comparability with other studies.

Two Categories Classification

In the first attempt to classify the dataset, the dataset was segmented into two categories: damaged roads and clean roads. Using these two categories, the pre-trained networks were repurposed through transfer learning and then tested to assess the viability of this classification. The classification accuracy results obtained from all the networks exceeded 99%, indicating that the process for transfer learning pre-trained networks to detect damaged roads is viable. Figure 66 highlights a sample of the classification results using this two-category classifier.



Figure 66. Photos. Two category neural network classification output of disaster damaged roads

Six Categories Classification

After validating the viability of the process, the dataset was segmented into six different classes: Damaged Roads, Clear Roads, Blocked Roads, Boats in Roads, Fallen Power Lines, and Flooded Roads. As previously mentioned, a total of three different CNNs (AlexNet, GoogLeNet, ResNet50) were investigated. The mean F1 score, recall, specificity, and precision were calculated after applying the four-fold cross-validation. The classification quality measure results for each network are presented in Tables 10, 11, and 12.

Table 10. AlexNet classification results

Network Accuracy: 74.1%				
Classes	Precision	Recall	F1	Specificity
Blocked Rd	0.375	0.300	0.333	0.847
Boat in Rd	1.000	1.000	1.000	1.000
Clear Rd	1.000	1.000	1.000	1.000
Damaged Rd	0.800	0.400	0.533	0.977
Flooded Rd	0.818	1.000	0.899	0.955
Power Lines	0.600	0.900	0.720	0.864

Table 11. GoogLeNet classification results

Network Accuracy: 68.5%				
Classes	Precision	Recall	F1	Specificity
Blocked Rd	0.359	0.225	0.277	0.886
Boat in Rd	0.935	0.975	0.955	0.945
Clear Rd	0.845	1.000	0.916	0.979
Damaged Rd	0.609	0.500	0.549	0.909
Flooded Rd	0.731	0.527	0.613	0.967
Power Lines	0.568	0.925	0.704	0.818

Table 12. ResNet50 classification results

Network Accuracy: 70.4%				
Classes	Precision	Recall	F1	Specificity
Blocked Rd	0.338	0.125	0.182	0.932
Boat in Rd	0.955	1.000	0.977	0.977
Clear Rd	0.803	1.000	0.891	0.959
Damaged Rd	0.733	0.550	0.628	0.977
Flooded Rd	0.866	0.723	0.788	0.977
Power Lines	0.534	0.975	0.690	0.818

The best performing network in terms of average accuracy was AlexNet, with an achieved accuracy of 74.1%. Additionally, AlexNet was the fastest network to train, thus requiring less computational power. Even though AlexNet produced the highest accuracy in this experiment, the more complex CNNs, such as GoogLeNet and ResNet50, would have provided much higher classification accuracies with larger training and testing datasets.⁽³⁶⁾ The fact that the image library had a relatively small number of images (especially after segmenting it into six categories) limited the ability of these neural networks to converge, which resulted in lower accuracies.

When assessing the Recall of each category across all CNNs tested, it becomes evident from the recall measurements obtained in the Blocked Road and Damaged Road categories that the networks need additional training data. These measurements show that the networks are getting confused between both these categories, thus lowering the overall accuracy of the network. Extending the database would show clear improvement that would get the classification closer to actual accuracy.

It is also notable that the Precision across the networks also showed that the Blocked Roads and Damaged Roads produced low measurements. The percentage of positive identifications that were correct was low. This again shows that the networks need more training with more images for these classes to learn more features and correctly differentiate between them.

Figures 67, 68, and 69 display the confusion matrices for AlexNet, GoogLeNet, and ResNet 50, respectively. The blocked road category was the least accurately classified category across all three networks, mainly due to the small size of the dataset.

Confusion Matrix

Output Class	Blocked Road	3 5.6%	0 0.0%	0 0.0%	4 7.4%	0 0.0%	1 1.9%	37.5% 62.5%
	Boat Road	0 0.0%	10 18.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Clear Road	0 0.0%	0 0.0%	5 9.3%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Damaged Road	1 1.9%	0 0.0%	0 0.0%	4 7.4%	0 0.0%	0 0.0%	80.0% 20.0%
	Flooded Road	0 0.0%	0 0.0%	0 0.0%	2 3.7%	9 16.7%	0 0.0%	81.8% 18.2%
	Power Lines	6 11.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	9 16.7%	60.0% 40.0%
		30.0% 70.0%	100% 0.0%	100% 0.0%	40.0% 60.0%	100% 0.0%	90.0% 10.0%	74.1% 25.9%
	Target Class							
	Blocked Road	Boat Road	Clear Road	Damaged Road	Flooded Road	Power Lines		

Figure 67. Graph. AlexNet confusion matrix

Confusion Matrix

Output Class	Blocked Road	1 1.9%	0 0.0%	0 0.0%	1 1.9%	0 0.0%	0 0.0%	50.0% 50.0%
	Boat Road	1 1.9%	10 18.5%	0 0.0%	0 0.0%	1 1.9%	0 0.0%	83.3% 16.7%
	Clear Road	0 0.0%	0 0.0%	5 9.3%	0 0.0%	1 1.9%	0 0.0%	83.3% 16.7%
	Damaged Road	0 0.0%	0 0.0%	0 0.0%	8 14.8%	4 7.4%	0 0.0%	66.7% 33.3%
	Flooded Road	0 0.0%	0 0.0%	0 0.0%	1 1.9%	3 5.6%	0 0.0%	75.0% 25.0%
	Power Lines	8 14.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	10 18.5%	55.6% 44.4%
		10.0% 90.0%	100% 0.0%	100% 0.0%	80.0% 20.0%	33.3% 66.7%	100% 0.0%	68.5% 31.5%
	Blocked Road	Boat Road	Clear Road	Damaged Road	Flooded Road	Power Lines		
	Target Class							

Figure 68. Graph. GoogLeNet confusion matrix

Confusion Matrix

Output Class	Blocked Road	2 3.7%	0 0.0%	0 0.0%	3 5.6%	0 0.0%	0 0.0%	40.0% 60.0%
	Boat Road	1 1.9%	10 18.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	90.9% 9.1%
	Clear Road	0 0.0%	0 0.0%	5 9.3%	1 1.9%	1 1.9%	0 0.0%	71.4% 28.6%
	Damaged Road	0 0.0%	0 0.0%	0 0.0%	4 7.4%	1 1.9%	0 0.0%	80.0% 20.0%
	Flooded Road	0 0.0%	0 0.0%	0 0.0%	1 1.9%	7 13.0%	0 0.0%	87.5% 12.5%
	Power Lines	7 13.0%	0 0.0%	0 0.0%	1 1.9%	0 0.0%	10 18.5%	55.6% 44.4%
		20.0% 80.0%	100% 0.0%	100% 0.0%	40.0% 60.0%	77.8% 22.2%	100% 0.0%	70.4% 29.6%
	Blocked Road	Boat Road	Clear Road	Damaged Road	Flooded Road	Power Lines		
	Target Class							

Figure 69. Graph. ResNet 50 confusion matrix

CHAPTER 6. CONCLUSION AND RECOMMENDATIONS

CONCLUSION

An unmanned aerial vehicle-based automated disaster assessment system was developed to assess damage caused by natural disasters. This system captures bird's eye view high-definition videos of disaster impacted areas. These videos are then sent to a ground station where they are sampled and classified, in real-time, using Convolutional Neural Networks (CNNs). This system also automatically tags the classified disaster-damaged roads on an ArcGIS map to provide real-time geolocation feedback of the damage caused by the disaster. Moreover, the system provides live streaming of the UAV's video feed to an RTMP server, enabling the first responders to assess damages.

To develop this system, a detailed comparative analysis of some of the leading commercial and industrial UAV platforms were conducted. This analysis provided a list of recommendations to different UAV platforms that are good candidates for this project. However, the final UAV platform recommendation for this project was the DJI Matrice 300 RTK. The analysis concluded that the new DJI Matrice 300 RTK exceeds the expectations compared to the DJI Matrice 600 Pro in key areas such as wind speed tolerance, range, speed, flight time, and reliability. Furthermore, the Matrice 300 RTK applies newer AI-based technology such as Smart Track and Live Mission Recording, enabling the user to have an automated system that returns accurate data for mission completion. The key reason this platform was selected as the final recommendation is its impressive ability to resist wind speeds up to 54 kph or 15 m/s, which surpassed all other UAV platforms analyzed.

A customized GUI application was developed using Python and MATLAB software to automate and centralize the operation of this system. The application included managing, sampling, classifying, and ArcGIS map tagging of the UAV-generated video streams. This application also provided some flexibility to customize the operating settings of this system.

Due to the particular nature of this application, field tests were not viable. Therefore, the system was extensively simulated and tested using a compiled library of images captured from previous natural disasters. The image library was classified into six categories: damaged roads, clear roads, blocked roads, boats in roads, fallen power lines, and flooded roads. A total of three different CNN (AlexNet, GoogLeNet, ResNet50) classifiers were investigated after applying transfer learning and utilizing four-fold cross-validation. AlexNet achieved the highest accuracy of 74.1%. Even though AlexNet produced the highest accuracy in this experiment, the more complex CNNs such as GoogLeNet and ResNet50 would have provided much higher classification accuracies given larger training and testing datasets.

RECOMMENDATIONS FOR FUTURE WORK

The major area that could be improved is the image library. The fact that this library had a relatively small size impacted the classification accuracy. Not only this, but also since this library is compiled of images of previous natural disasters, there was no consistency in the quality of images, which also impacts the classification accuracy. Therefore, the image library could be improved through the utilization of the developed system to gather higher quality and consistent images. Eventually, these newly collected images could be used to retrain the classifier to improve its accuracy. Finally, after this system has been in use for a while, the GUI application could be further customized to include more features that streamline the functionality of this system.

REFERENCES

1. NOAA, National Centers for Environmental Information (NCEI), “U.S. Billion-Dollar Weather and Climate Disasters,” NOAA, 2021, <https://www.ncdc.noaa.gov/billions> , DOI: 10.25921/stkw-7w73.
2. IFRC, “World Disasters Report 2020: Come Heat or High Water - Tackling the Humanitarian Impacts of the Climate Crisis Together [EN/AR],” ReliefWeb, accessed September 17, 2021, <https://reliefweb.int/report/world/world-disasters-report-2020-come-heat-or-high-water-tackling-humanitarian-impacts>.
3. Tetila, E. C., B. B. Machado, G. K. Menezes, A. D. S. Oliveira, M. Alvarez, W. P. Amorim, N. A. D. S. Belete, G. G. D. Silva, H. Pistori, “Automatic Recognition of Soybean Leaf Diseases Using UAV Images and Deep Convolutional Neural Networks,” *IEEE Geoscience and Remote Sensing Letters* 17, no. 5 (2019): 903–907.
4. Comfort, L. K., K. Ko, and A. Zagorecki, “Coordination in Rapidly Evolving Disaster Response Systems: The Role of Information,” *American Behavioral Scientist* 48, no. 3 (2004): 295–313.
5. Zhu, Y. J., Y. Hu, and J. M. Collins, “Estimating Road Network Accessibility During a Hurricane Evacuation: A Case Study of Hurricane Irma in Florida,” *Transportation Research Part D: Transport and Environment* 83 (2020): 102334.
6. LeCun, Y., and Y. Bengio, “Convolutional Networks for Images, Speech, and Time Series,” *The Handbook of Brain Theory and Neural Networks* 3361, no. 10 (1995).
7. Zeiler, M. D., and R. Fergus, “Visualizing and Understanding Convolutional Networks,” in *European Conference on Computer Vision* (Cham: Springer, September, 2014), 818–833.
8. Prabhu, “CNN Architectures - LeNet, Alexnet, VGG, GoogLeNet and ResNet,” Medium, March 15, 2018, <https://medium.com/@RaghavPrabhu/cnn-architectures-lenet-alexnet-vgg-googlenet-and-resnet-7c81c017b848>.
9. Whatmough, P. N., C. Zhou, P. Hansen, S. K. Venkataramanaiah, J. S. Seo, and M. Mattina, “Fixynn: Efficient Hardware for Mobile Computer Vision via Transfer Learning,” *arXiv Preprint* 1902 (2019): 11128.
10. Krizhevsky, A., I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Communication of ACM* 60, no. 6 (June, 2017): 84–90.
11. Dwivedi, P., “Understanding and Coding a ResNet in Keras,” Medium, March 27, 2019, <https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33>.

12. NCDOT, "Project: Improved Approaches to Environmental Compliance During Highway Construction," TRID Database, 2018, retrieved September 1, 2021, <https://connect.ncdot.gov/projects/research/RNAProjDocs/RP2019-05%20Final%20Report.pdf>.
13. CTECH, "Project: Tracking Shoreline Conditions to Protect," TRID Database, 2018, retrieved September 1, 2021.
14. MPC, "Project: Development of Unmanned Aerial Vehicle (UAV) Bridge Inspection Procedures," TRID Database, 2017, retrieved September 1, 2021.
15. TCSCS, "Project: Bridge Inspecting with Unmanned Aerial Vehicles R&D," TRID Database, 2017, retrieved September 1, 2021.
16. TRB, "Project: Railroad Bridge Inspections for Maintenance and Replacement Prioritization Using Unmanned Aerial (UAVs) with Laser Scanning Capabilities," TRID Database, 2016, retrieved September 1, 2021.
17. Bupe, P., R. J. Haddad, and F. Rios, "Relief and Emergency Communication Network Based on an Autonomous Decentralized UAV Clustering Network," *IEEE SoutheastCon*, 2015.
18. Ro, A., J. Oh, and L. Dong, "Lessons Learned: Application of Small UAV for Urban Highway Traffic Monitoring," *45th AIAA Aerospace Sciences Meeting, and Exhibit, Aerospace Sciences Meetings*, 2007.
19. U.S. Army, "Eyes of the Army": U.S. Army Roadmap for UAS 2010–2035, <https://irp.fas.org/program/collect/uas-army.pdf>.
20. Federal Aviation Administration, "Electronic Code of Federal Regulations: Title 14: Chapter I: Subchapter F: Part 107: Small Unmanned Aircraft Systems", Sept. 2021, [online] Available: <https://www.ecfr.gov/current/title-14/chapter-I/subchapter-F/part-107?toc=1>.
21. Stewart, S. R., and R. Berg, "National Hurricane Center Tropical Cyclone Report Hurricane Florence (AI062018)," National Hurricane Center, 2019, p. 9.
22. DJI, "Zenmuse H20 Series – Unleash the Power of One," DJI, n.d., accessed September 2, 2021, <https://www.dji.com/zenmuse-h20-series>.
23. DJI, "Matrice 600 Pro," DJI, accessed September 2, 2021, <https://www.dji.com/matrice600-pro>.
24. DJI, "Matrice 300 RTK," DJI, accessed September 2, 2021, <https://www.dji.com/matrice-300>.
25. DJI, "Phantom 4 RTK," DJI, accessed September 2, 2021, <https://www.dji.com/phantom-4-rtk>.

26. Yuneec, “Yuneec H520 RTK,” Yuneec, accessed September 2, 2021, <https://us.yuneec.com/h520-series/>.
27. Alta, “Freefly Alta 8 Pro,” Alta, accessed September 2, 2021, <https://freeflysystems.com/alta-8>.
28. DJI, “Mavic 2 Pro,” DJI, accessed September 2, 2021, <https://www.dji.com/mavic-2>
29. DJI, “Matrice 210 V2 RTK,” DJI, accessed September 2, 2021, <https://www.dji.com/matrice-200-series-v2>.
30. Mariluz, O., “Earthquake in Loreto: What Roads in the Country are Blocked or Restricted after the Earthquake?” RPP, May 27, 2019, accessed September 8, 2021, <https://rpp.pe/peru/loreto/terremoto-en-loreto-que-carreteras-del-pais-estan-bloqueadas-o-con-paso-restringido-tras-el-sismo-noticia-1199255>.
31. Sisson, P. “How the Country’s Deadliest City for Pedestrians Plans to Save Lives with Safer Streets,” *Curbed*, January 22, 2020, accessed September 8, 2021, <https://archive.curbed.com/2020/1/22/21064325/orlando-crash-cycling-pedestrian-traffic-safety>.
32. Shelton, C., “Waverly Crews Rushing to Remove Debris, Distribute Tarps ahead of Ida Remnants Reaching TN,” *WZTV*, August 30, 2021, accessed September 8, 2021, <https://fox17.com/news/local/waverly-crews-rushing-to-remove-debris-distribute-tarps-ahead-of-ida-remnants-reaching-tn>.
33. Telegraph Media Group, “Japan Earthquake: 30 Pictures of Boats and Ships Swept Ashore by the Tsunami,” *The Telegraph*, March 18, 2011, accessed September 18, 2021, <https://www.telegraph.co.uk/news/picturegalleries/worldnews/8390718/Japan-earthquake-30-pictures-of-boats-and-ships-swept-ashore-by-the-tsunami.html?image=14>.
34. Everton Bailey Jr., “Heavy Rain, Winds Leave Thousands of Oregonians without Power,” *Oregonlive*, November 20, 2012, accessed September 18, 2021, https://www.oregonlive.com/pacific-northwest-news/2012/11/heavy_rain_winds_leave_thousan.html.
35. NBC, “Flooding Shuts Down Darlington Bridge until Early Sat., Closes Schools,” *NBC online*, accessed September 18, 2021, <https://www.nbc15.com/content/news/Flooding-closes-Main-Street-bridge-schools-in-Darlington-562054731.html>.
36. Bocanegra, M. G., and R. J. Haddad, “Convolutional Neural Network-Based Disaster Assessment Using Unmanned Aerial Vehicles,” *SoutheastCon* (2021): 1–6.