Project 2226 | November 2023

SJSU SAN JOSÉ STATE UNIVERSITY

MTI MINETA TRANSPORTATION INSTITUTE

AI-Based Bridge and Road Inspection Framework Using Drones

Dr. Hovannes Kulhandjian

CSUTC California State University Transportation Consortium

FRESNO STATE
Discovery. Diversity. Distinction.

MINETA TRANSPORTATION INSTITUTE

Founded in 1991, the Mineta Transportation Institute (MTI), an organized research and training unit in partnership with the Lucas College and Graduate School of Business at San José State University (SJSU), increases mobility for all by improving the safety, efficiency, accessibility, and convenience of our nation's transportation system. Through research, education, workforce development, and technology transfer, we help create a connected world. MTI leads the Mineta Consortium for Transportation Mobility (MCTM) and the Mineta Consortium for Equitable, Efficient, and Sustainable Transportation (MCEEST) funded by the U.S. Department of Transportation, the California State University Transportation Consortium (CSUTC) funded by the State of California through Senate Bill 1 and the Climate Change and Extreme Events Training and Research (CCEETR) Program funded by the Federal Railroad Administration. MTI focuses on three primary responsibilities:

Research

MTI conducts multi-disciplinary research focused on surface transportation that contributes to effective decision making. Research areas include: active transportation; planning and policy; security and counterterrorism; sustainable transportation and land use; transit and passenger rail; transportation engineering; transportation finance; transportation technology; and workforce and labor. MTI research publications undergo expert peer review to ensure the quality of the research.

Education and Workforce Development

To ensure the efficient movement of people and products, we must prepare a new cohort of transportation professionals who are ready to lead a more diverse, inclusive, and equitable transportation industry. To help achieve this, MTI sponsors a suite of workforce development and education opportunities. The Institute supports educational programs offered by the Lucas Graduate School of Business: a Master of Science in Transportation Management, plus graduate certificates that include High-Speed and Intercity Rail Management and Transportation Security Management. These flexible programs offer live online classes so that working transportation professionals can pursue an advanced degree regardless of their location.

Information and Technology Transfer

MTI utilizes a diverse array of dissemination methods and media to ensure research results reach those responsible for managing change. These methods include publication, seminars, workshops, websites, social media, webinars, and other technology transfer mechanisms. Additionally, MTI promotes the availability of completed research to professional organizations and works to integrate the research findings into the graduate education program. MTI's extensive collection of transportation-related publications is integrated into San José State University's world-class Martin Luther King, Jr. Library.

Report 23-28

AI-Based Bridge and Road Inspection Framework Using Drones

Dr. Hovannes Kulhandjian

November 2023

A publication of the
Mineta Transportation Institute
Created by Congress in 1991

College of Business
San José State University
San José, CA 95192-0219

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. 23-28	**2. Government Accession No.**	**3. Recipient's Catalog No.**	
4. Title and Subtitle AI-Based Bridge and Road Inspection Framework Using Drones		**5. Report Date** November 2023	
		6. Performing Organization Code	
7. Authors Hovannes Kulhandjian, PhD https://orcid.org/0000-0002-8983-6757		**8. Performing Organization Report** CA-MTI-2226	
9. Performing Organization Name and Address Mineta Transportation Institute College of Business San José State University San José, CA 95192-0219		**10. Work Unit No.**	
		11. Contract or Grant No. ZSB12017-SJAUX	
12. Sponsoring Agency Name and Address State of California SB1 2017/2018 Trustees of the California State University Sponsored Programs Administration 401 Golden Shore, 5th Floor Long Beach, CA 90802		**13. Type of Report and Period Covered**	
		14. Sponsoring Agency Code	
15. Supplemental Notes DOI: 10.31979/mti.2023.2226			
16. Abstract There are over 590,000 bridges dispersed across the roadway network that stretches across the United States alone. Each bridge with a length of 20 feet or greater must be inspected at least once every 24 months, according to the Federal Highway Act (FHWA) of 1968. This research developed an artificial intelligence (AI)-based framework for bridge and road inspection using drones with multiple sensors collecting capabilities. It is not sufficient to conduct inspections of bridges and roads using cameras alone, so the research team utilized an infrared (IR) camera along with a high-resolution optical camera. In many instances, the IR camera can provide more details to the interior structural damages of a bridge or a road surface than an optical camera, which is more suitable for inspecting damages on the surface of a bridge or a road. In addition, the drone inspection system is equipped with a minicomputer that runs Machine Learning algorithms. These algorithms enable autonomous drone navigation, image capture of the bridge or road structure, and analysis of the images. Whenever any damage is detected, the location coordinates are saved. Thus, the drone can self-operate and carry out the inspection process using advanced AI algorithms developed by the research team. The experimental results reveal the system can detect potholes with an average accuracy of 84.62% using the visible light camera and 95.12% using a thermal camera. This developed bridge and road inspection framework can save time, money, and lives by automating and having drones conduct major inspection operations in place of humans.			
17. Key Words Bridge inspection, Road inspection, Applied machine learning, Autonomous navigation, Drones	**18. Distribution Statement** No restrictions. This document is available to the public through The National Technical Information Service, Springfield, VA 22161.		
19. Security Classif. (of this report) Unclassified	**20. Security Classif. (of this page)** Unclassified	**21. No. of Pages** 47	**22. Price**

Form DOT F 1700.7 (8-72)

DOI: 10.31979/mti.2023.2226

Mineta Transportation Institute
College of Business
San José State University
San José, CA 95192-0219

Tel: (408) 924-7560
Fax: (408) 924-7565
Email: mineta-institute@sjsu.edu

transweb.sjsu.edu/research.2226

ACKNOWLEDGMENTS

This study was supported by both the CSU Transportation Consortium and the Fresno State Transportation Institute. Any opinions, findings, conclusions, and recommendations expressed in this material are those of the author and do not necessarily reflect the views of these institutes. The author would like to thank his student assistants, Jose Maciel Torres, Cruz Nieves, Christian Reeves, and Nicholas Amely, undergraduate students from the Electrical and Computer Engineering (ECE) department at California State University, Fresno for working on this interesting research project and implementing and testing the final proposed prototype. The author would also like to thank Editing Press, for editorial services, as well as MTI staff, including Executive Director Karen Philbrick, PhD; Deputy Executive Director Hilary Nixon, PhD; Graphic Designer Alverina Eka Weinardy; and Executive Administrative Assistant Jill Carter.

TABLE OF CONTENTS

LIST OF FIGURES

Executive Summary

Bridge and road inspections are a necessary part of maintaining the country's infrastructure. However, they can be time-consuming and expensive. Drones offer a new way to inspect bridges and roads that is faster, safer, and more cost-effective than traditional methods. Drones can access areas that are difficult or dangerous for humans, such as under bridges or along train tracks. They can also collect data from a variety of angles, which can help to identify problems that would not be visible from a single vantage point. In addition to being more efficient, drones are also safer than traditional inspection methods. This is because drones can fly over bridges without being close to traffic lanes or putting inspectors in harm's way. While the use of drones for bridge inspections is still in its early stages, it has the potential to revolutionize the way we inspect our bridges. Furthermore, drone inspections can reduce the environmental impact of bridge and road inspections by reducing the need for traffic control and other measures that can disrupt traffic and pollute the air. As the technology continues to develop, drones will become even more valuable for bridge and road inspection.

Future bridge and road inspections will benefit substantially from our proposed system. In this study, we developed and implemented a framework based on artificial intelligence (AI) for inspecting bridges and roads using drones equipped with a variety of sensors and a minicomputer. We use an infrared (IR) camera coupled with a high-resolution optical camera since using optical cameras alone is insufficient. When compared to an optical camera, which is more suited for inspecting damage on the surface of a bridge or a road, the IR camera frequently provides more information on the interior structural faults of a bridge or road. To enable autonomous drone navigation and the capture of photographs of the bridge or road structure whenever it detects any problems, our drone inspection system is fitted with a minicomputer running Machine Learning algorithms. Using sophisticated AI algorithms, the drone both self-operates and does the inspection procedure without human assistance. Experiments were conducted on roads, not bridges, due to the stricter regulations of flying a drone near a bridge. The experimental results revealed that the system can detect potholes with an average accuracy of 84.6% using the visible light camera and 95.1% using an IR camera. We believe this bridge and road inspection framework can save considerable time, money, and lives by automating, and having drones conduct, major inspection operations in place of humans.

1. Introduction

There are over 590,000 bridges dispersed across the roadway network stretching across the United States alone. Each bridge that has a length of 20 feet or greater must be inspected at least once every 24 months, according to the Federal Highway Act (FHWA) of 1968. Each inspection must adhere to the National Bridge Inspection Standards (NBIS), established by the US Department of Transportation (USDOT). A bridge inspection should uncover any severe structural flaws that need to be addressed, quantify the overall state of the bridge to prioritize capital needs, identify routine maintenance, and keep track of the bridge's history. Inspecting bridges is a time-consuming and expensive task. Traditional inspection methods require a lot of coordination, such as traffic control, and they put personnel in danger.

Several methods of routine bridge inspection are currently used by specialists to detect defects such as surface cracks or sub-surface delamination in infrastructure, including visible inspection, thermal imaging inspection, ground penetrating radar (GPR), and acoustic inspection.

For visible inspection, inspectors visually review the condition of the bridge in detail.

Thermal inspections can detect changes in infrared radiation from the surface of a bridge, which could indicate degradation or delamination in the concrete.

GPR uses electromagnetic radiation, inspectors can create an image of the area below the concrete in a bridge to find defects such as cracks or delamination in the materials.

Acoustic inspections are done using a hammer, chain drag, or some other tool, whereby inspectors listen for changes in sound pitch on the bridge. Acoustic testing can be used to detect splits or separations in the materials used to make the bridge, such as delamination or coating splits.

Regardless of the technique, the current process requires complex traffic lane-closure management, additional labor hours, expensive equipment, and can potentially place workers in unsafe environments (Flyability, 2022).

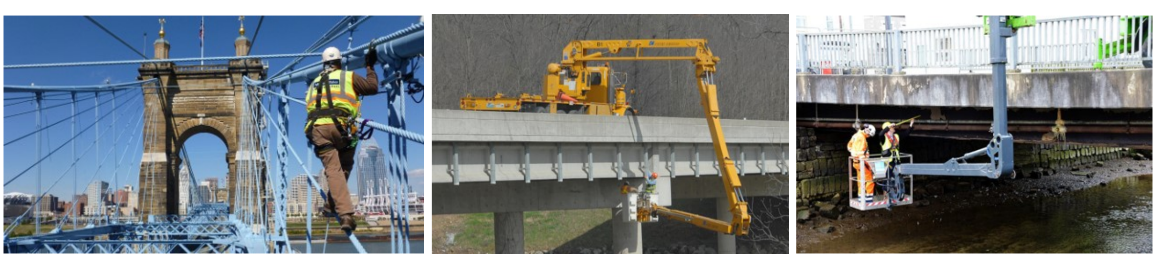

Figure 1. Sample Images of Bridge Inspections Conducted by Specialists (Cho, 2018)

As can be seen by the sample images in Figure 1, bridge inspections can be a tedious and dangerous task that might take several weeks to complete.

Figure 2. Inspection Methods Used for Bridge and Road Inspection Include
Ground Penetrating Radar, a Chain Drag and Other Tools (USDT-FHWT)

Figure 2 shows different methods that are used in bridge inspection, including GPR, chain drag, as well as other tools like a hammer.

Drones, on the other hand, can readily access regions that humans find difficult or dangerous, such as under bridges or along train tracks. They allow workers to keep a longer standoff distance while still collecting the data required for inspections. In comparison to traditional inspection equipment like snoopers (truck equipment), drones can capture significantly more thorough inspection data. They make collecting high-definition photos from limited and inaccessible areas, such as beneath bridges and along beams and girders, simple. During bridge inspections, drones have the potential to save costs, deliver better data, and increase worker safety. Drone inspections of bridges will considerably reduce inspection expenses, as noted by Minnesota Department of Transportation (MnDOT), which was involved in research focusing on using drones as a tool for increasing the quality of bridge inspections: a normal bridge inspection requires three snooper inspection vehicles and eight inspection days; on average that can cost $59,000 (Wells & Lovelace, 2017). According to a report published by the United States Department of Transportation, Office of the Assistant Secretary for Research and Technology, "The use of drones for bridge inspections can create an overall average cost savings of 40 percent without a reduction in inspection quality" (USDOT-OASRT, 2019).

Recently, on Jan. 28, 2022, the Fern Hollow bridge in Pittsburgh collapsed. The National Transportation Safety Board (NTSB) reported that there might have been some structural damage that was not detected (NTSB, 2023). Had the bridge been inspected with a drone, that structural damage might have been detected and such a devastating collapse could have been avoided.

Our proposed research discussed below will be of great benefit for future bridge and road inspections. In this research work, we developed an artificial intelligence (AI)-based hardware and software framework for bridge and road inspection using drones with multiple sensors. Since it is insufficient to conduct inspections using cameras alone, we utilized an infrared (IR) camera along with a high-resolution optical camera. In many instances, the IR camera can provide more details regarding the interior structural damage of a bridge or a road than an optical camera, which is more suitable for inspecting damage on the surface of a bridge or a road.

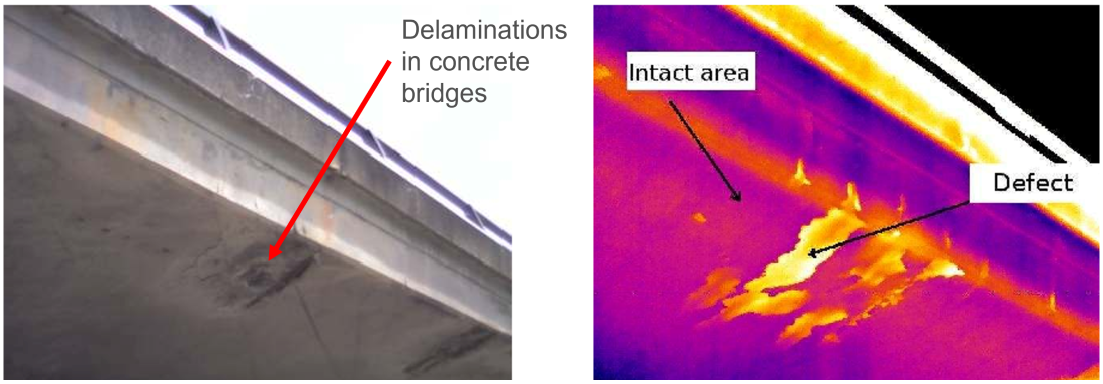

Figure 3. Example Images of a Bridge Taken with a Visible Camera (left)
as well as an Infrared Camera (right)

Figure 3 shows example images of a bridge taken with a visible camera as well as an infrared camera. The image on the left, taken with the visible camera, does not show all the details of the damage underneath the concrete structure, whereas it can be clearly seen using the infrared camera, as shown on the right.

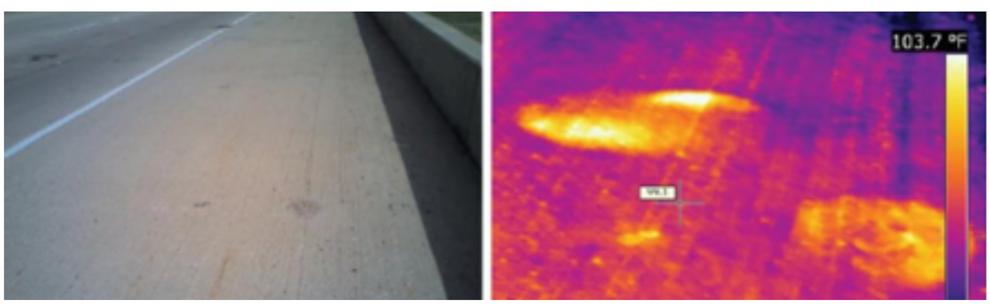

Figure 4. Example Images of a Road Surface Taken with a Visible Camera
as well as an Infrared Camera

Similarly, Figure 4 shows example images of a road surface, in this case a concrete road, that were taken with a visible camera as well as an infrared camera. As can be seen, the image on the left, taken with the visible camera, does not show all the details of the damage that is detected underneath the concrete structure, whereas the damage can be clearly seen in the image on the right, taken with the infrared camera.

Therefore, we believe that the integration of an infrared camera along with a visible camera can greatly enhance the performance of fault detection on bridges as well as on roads.

Several research studies have explored the use of drones to investigate bridges and other structures. Using ground penetrating radar, Biscarini et al. (2020) investigated the use of unmanned aerial photogrammetric surveys to perform visual inspections and produce a geometrical 3D model. Infrared thermography analyses were also carried out to characterize the thermal surface map of the structure and to detect anomalies related to material degradation, such as the presence of humidity.

Another study, Seo & Wacker (2018), evaluates the capabilities of drone technology as a supplemental bridge inspection tool to support legally mandated conventional bridge inspections. A visible light camera was used in this study but no infrared camera.

The Minnesota Department of Transportation and Collins Engineers have been investigating the use of unmanned aircraft systems (UASs) as a tool for bridge inspections in a multi-phase project, according to Wells & Lovelace's (2017) report. The MnDOT's earlier study was carried out in this round of research, which also highlighted potential uses of UAS technology to support bridge inspections.

In all the prior studies mentioned, advanced machine learning algorithms were not explored and there was a need for a specialist to operate the drone. Conversely, our drone inspection system is equipped with a minicomputer that runs Machine Learning algorithms to enable autonomous drone navigation and to take images of the bridge or the road structure on the fly; whenever it detects any damage, it can save the location information of that damage. Instead of having a person operate the drone, it can self-operate and carry out the inspection process on its own using the advanced AI algorithms we have developed. In this research work, we focused our experimental results on road inspection since getting permission to scan bridges was not as simple as we expected. The proposed framework can also be used for performing bridge inspection, which we may explore in our future work.

2. System Overview

The AI-based bridge and road inspection system uses advanced software, a machine-learning neural network, and hardware subsystems that all work together to perform the intended task.

Within the software subsystem, software development kit (SDK) libraries are used to interface with two different control setups. The libraries interface with the flight controller of the drone, which enables efficient communication between that device and the companion computer for preflight and inflight tests and data transmission. The libraries also interface with the detection sensors on board the drone, which are the cameras and ultrasonic sensors. This enables efficient video feed and image data transmission from the cameras to the companion computer for processing. Once the SDK libraries have proven a successful connection between the companion computer and the control setups, the drone's movement is initialized. The first step in the software's process is to instruct the drone to take off to preset global positioning system (GPS) waypoints. As the drone travels to these waypoints, it takes continuous pictures of the road that it is flying over. These images are analyzed in real time by the machine learning system, and if a fault is detected, the drone pauses its waypoint path and drops down in altitude to fly closer to the detected fault. Closeup pictures of the fault are taken by both cameras and transmitted to the ground station for storage, and then the drone resumes along its flight path and continues taking pictures. Once the drone reaches the final waypoint, the current mission is declared finished, and the drone returns to the launch waypoint.

Within the machine learning subsystem, multiple preflight processes are run. Once the model dataset is uploaded, the dataset is aggregated and then used to train the neural network. After the model is initially trained, it is subsequently employed in real-time to identify cracks in the images captured within the software subsystem.

All the hardware onboard the drone is encompassed within the hardware subsystem. This includes the unmanned aerial vehicle (UAV) hardware, the machine learning hardware, and the object avoidance hardware. All the image sensors within the machine learning and object avoidance hardware sections are tested preflight, and if all the sensors can have interpretable data drawn from them, the Electronic Speed Controllers (ESCs) are calibrated, and the flight controller is armed. The software subsystem processes and the detection portion of the machine learning subsystem are then started.

The AI-bridge and road inspection system flowchart is shown in Figure 5.

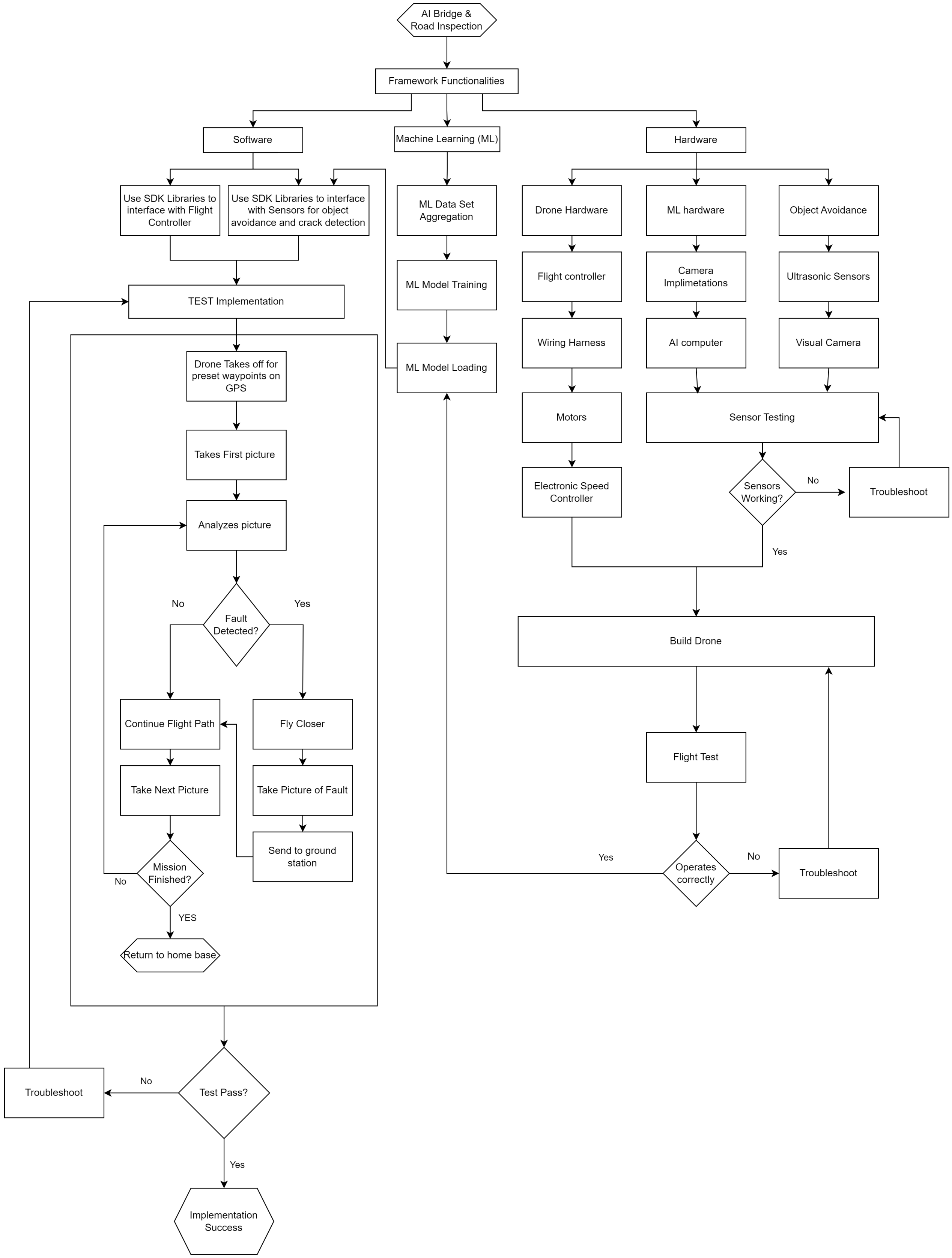

Figure 5. The AI-Bridge and Road Inspection System Flowchart

3. Drone Framework

We built our system on a Hexacopter 6-axle Aircraft Kit that has an HMF S550 Frame, PXI PX4 Flight Control, 920KV Motors, a GPS, and an AT9 Transmitter.

The PXI PX4 Flight Controller, shown in Figure 6, is an open-source autopilot system that can be used to control a variety of UAVs. It is based on the PX4 autopilot software, which is developed and maintained by a community of developers from around the world. The PXI PX4 Flight Control is designed to be modular and scalable, making it possible to use it for a wide range of UAV applications.

Figure 6. PXI PX4 Pixhawk 2.4.8 Flight Controller

The PXI PX4 Flight Control consists of several different components, including a flight controller board, a power distribution board, a GPS receiver, an inertial measurement unit (IMU), and a variety of sensors. The flight controller board is the central processing unit of the system and is responsible for controlling the UAV's flight. The power distribution board provides power to the other components of the system. The GPS receiver provides the flight controller with information about the UAV's position and altitude. The IMU provides the flight controller with information about the UAV's orientation. The sensors provide the flight controller with information about the UAV's environment.

The PXI PX4 Flight Control can be programmed using the PX4 autopilot software. The software provides several different features, including:

- Flight control: Used to control the UAV's flight using a variety of different autopilot modes.
- Payload control: Used to control the UAV's payload, such as a camera or a gimbal.

- Mission planning: Used to plan and execute UAV missions.
- Data logging: Used to log data from the UAV's sensors.

The PXI PX4 Flight Control is a powerful and versatile autopilot system that can be used for a wide range of UAV applications. It is open-source, modular, and scalable, making it a good choice for both hobbyists and professional users.

Figure 7. The Final Drone Prototype

The final drone prototype, shown in Figure 7, including all the peripherals, six propellers with guards, a GPS module with built-in compass with GPS antenna mount and the PXI PX4 flight controller, is shown in Figure 7.

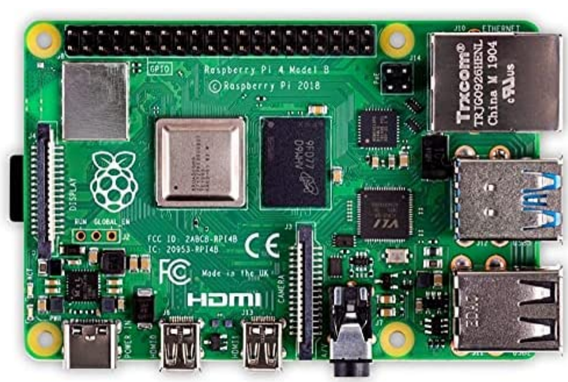

Figure 8. The Raspberry Pi 4B Minicomputer

The Raspberry Pi 4B minicomputer, shown in Figure 8, was embedded in the drone to perform all the command and control as well as data gathering and implementing all the machine learning algorithms.

Figure 9. Visible Light Camera

The visible light camera, shown in Figure 9, was installed on the drone, and was used for gathering the camera images.

Figure 10. Seek Thermal CompactPRO Infrared Camera

The Seek Thermal Camera, Seek Thermal CompactPRO, a High Resolution (320 x 240 thermal sensor) Thermal Imaging Camera, shown in Figure 10, was installed on the drone, and was used for gathering thermal infrared images. We selected this camera due to its high resolution and modest price tag.

Figure 11. Radiolink AT9S Pro 10/12 Channels Radio
Transmitter and Receiver R9DS

A Radiolink AT9S Pro 10/12 Channels Radio Transmitter and Receiver R9DS, shown in Figure 11, was used to communicate and control the drone. The transceiver is a high-performance radio system that is designed for use with a variety of remote-controlled aircraft, including drones, helicopters, and airplanes. The transmitter features a 10/12-channel design, which allows users to control a wide range of aircraft functions. The receiver is compatible with a variety of Radiolink receivers, including the R9DS, which is a 10-channel receiver that supports serial BUS and pulse width modulation (PWM) signals.

The S550 Hexacopter establishes completely wireless communication to the ground station via telemetry transmitters and receivers. The receivers were chosen with specific guidelines. The radio must have 2-way communication allowing it to send and receive telemetry specifications such as altitude, speed, location, and flight commands. The radios also needed to have a long communication distance so there was no chance of lost communication during a scheduled flight. The radios chosen that fit these requirements were the Holybro 915 MHz Telemetry Radios & the FrSky X8Rs Receiver, shown in Figure 12. The Holybro 915 MHz radios were chosen over a generic 5.8 GHz radio since the latter radio has a shorter range, uses a larger bandwidth, and draws more power from the energy-limited drone system. The 915 MHz radio has a smaller bandwidth, less power draw, a longer range for communication, and is unlikely to be interfered with by other frequency-communicating devices, as the 5.8 GHz radios are prone to be.

Figure 12. Telemetry Transmitter and Receiver

The FrSky X8Rs receiver, shown in Figure 13, is a compact Serial Bus (SBUS) protocol receiver that works with the Radio Master Tx16s radio controller. The X8Rs are sometimes provided with a non-US software framework and must be flashed with the US framework to work with specific equipment.

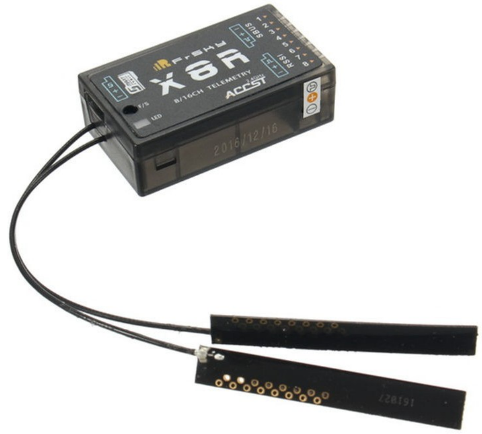

Figure 13. Drone Receiver for Radio Master Tx16s Transmitter

These radios are connected to the drone's flight controller (Pixhawk). The Pixhawk is integrated with a user-friendly interface allowing users to tune, set up, and create specific flight guidelines that the Pixhawk listens to while in flight. On the drone there is a companion computer, the Raspberry Pi 4, that runs the entirety of the project's main purpose. Since the power drawn from the Pi when it is fully under the load of the scripted commands is roughly between 4.5 W – 5.5 W, there is no concern regarding a power loss mid-flight since the Raspberry Pi runs at a lower voltage, i.e., 5 V. Hence, a step-down voltage controller is used to drop down the voltage from the 11.1 V LiPo drone battery to 5 V. The step-down power system also operates the Pi's integrated visible light optical camera and the Seek Thermal Pro camera.

The electrical schematic diagram of the hardware framework is shown in Figure 14. On the left side, it shows the Seek Thermal Pro camera connected to the Raspberry Pi, as well as the visible camera. On the right side are the six propellers connected to the Pixhawk controller along with the GPS, the lipo battery, and the telemetry radios.

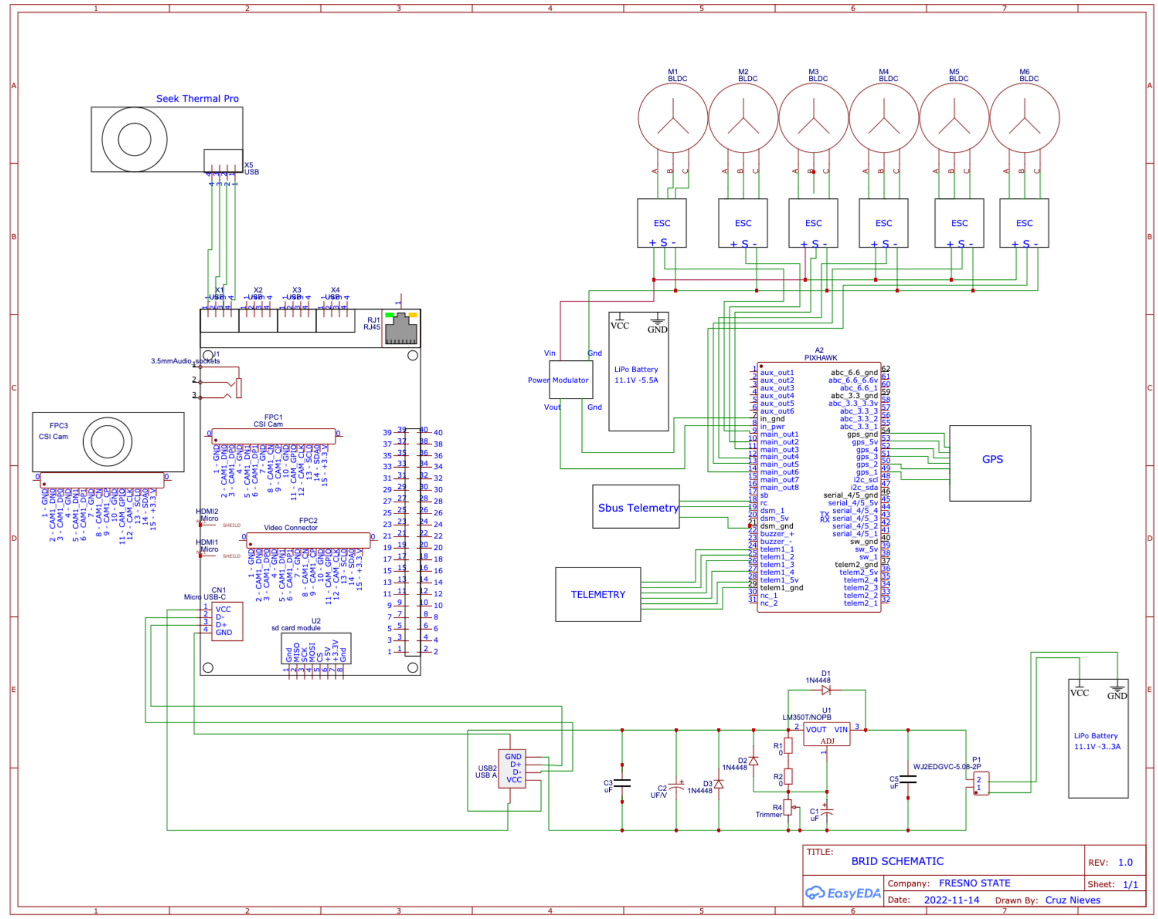

Figure 14. Electrical Schematic Diagram of the Hardware Framework

The overall system diagram is shown in Figure 15, comprised of the Raspberry Pi with the thermal and visible cameras connected to it. The Pixhawk controller has the Radio controller and the GPS interconnected to it. The Pixhawk module has complete control of the 6 propeller modules.

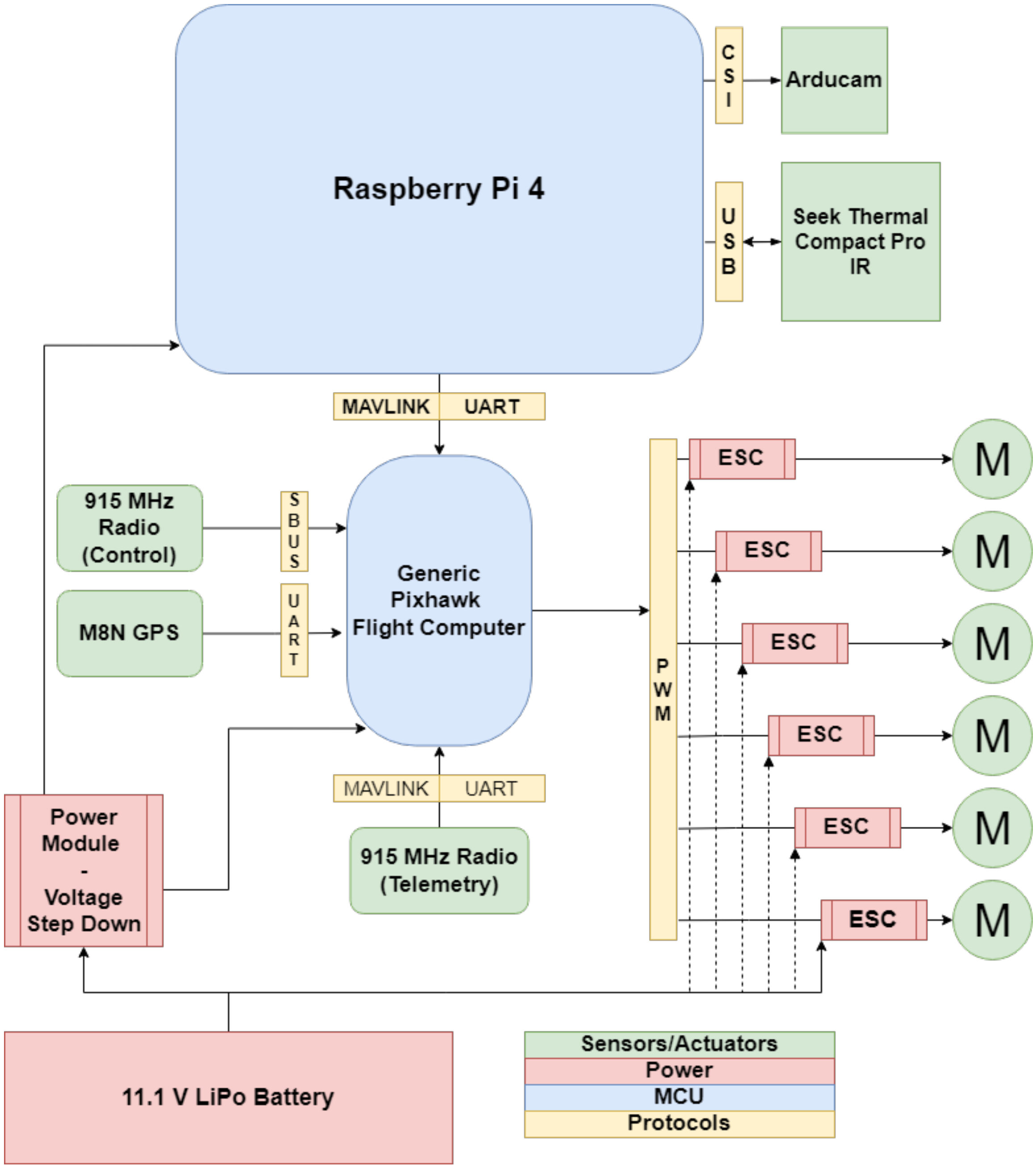

Figure 15. Drone System Diagram

4. Software Architecture

Our system is primarily built on a ROS2 (Robot Operating System) system architecture. ROS is a set of software libraries and tools for building robot applications. These libraries and algorithm implementations are state-of-the-art and are a common architecture that is used in industry. ROS primarily implements a version of a publisher-subscriber model known as nodes and topics, visually described by Figure 16. One benefit to publishing data from nodes as topics is that it allows for multiple nodes to use the data simultaneously, without the need to implement multi-threading; it is handled by ROS2 in the back end. A network of nodes and topics is formed for complex systems but can perform efficient, continuous growth and easy development by only building a specific node.

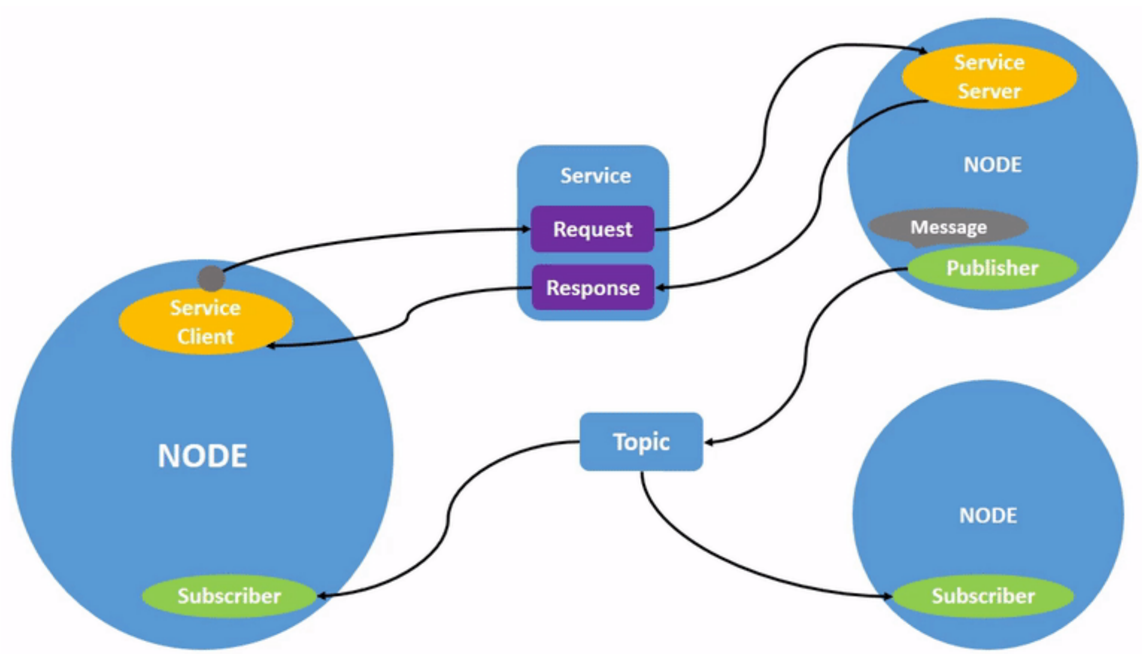

Figure 16. ROS2 Nodes and Topics

The ROS setup integrated into the project is divided into three different nodes that run on the Raspberry Pi companion computer: the Imaging Node, Road Nav Node, and Defect Classification Node. The three developed nodes are shown in Figure 17 and described below.

1. **Imaging Node**: The program that interfaces with physical cameras, processes the image with OpenCV and publishes the image topic as "sensor msg" type from an OpenCV Mat type. The Imaging node processes all the image and video data drawn from sensors onboard the drone. The Imaging node does not subscribe to any topics but publishes two different topics. Those two topics are 'thermal_st' and 'visible_light', which refer to the video feeds and image frames from the

thermal and visible light cameras onboard the drone. These frames are pulled from the cameras using multiple driver scripts and SDK libraries contained within the Imaging node.

2. **Road Nav Node**: Our motion, planning, and data collection program. Subscribes to the image topics, performs edge detection for autonomous road navigation based on planned GPS start and end points. Data collection is done at periodic intervals. The Road Nav node processes the Inertial Measurement Unit (IMU) and location data from multiple sensors onboard the drone. The Road_Nav node subscribes to three different topics but does not publish any topics itself. This node subscribes to the 'imu_pub' topic, which draws the IMU orientation, acceleration, and altitude data from it. It subscribes to the 'global_position' node topic, which draws GPS location data from it. Finally, it subscribes to the 'flagged' topic that is published by the Defect Classification node, which draws crack detection data from it. The node processes all the data from these three different topics and uses it to control the movement of the drone.

3. **Defect Classification Node**: The program that uses the developed machine learning model to detect defects from the subscribed image topics to publish the defect topic. The model's requirement for a Tensorflow lite format, plus delays and incompatibility with MATLAB, prevented full implementation of this feature. The Defect Classification node subscribes to two different topics and publishes one topic. The two topics it subscribes to are the two topics published by the Image node, which are 'thermal_st' and 'visible_light'. The node uses the video feed and image frame data to provide input into the machine learning neural networks and scripts contained within the node and publishes a 'flagged' topic that contains all the images that were classified as containing defects.

A visual representation of the above structure is shown in Figure 17.

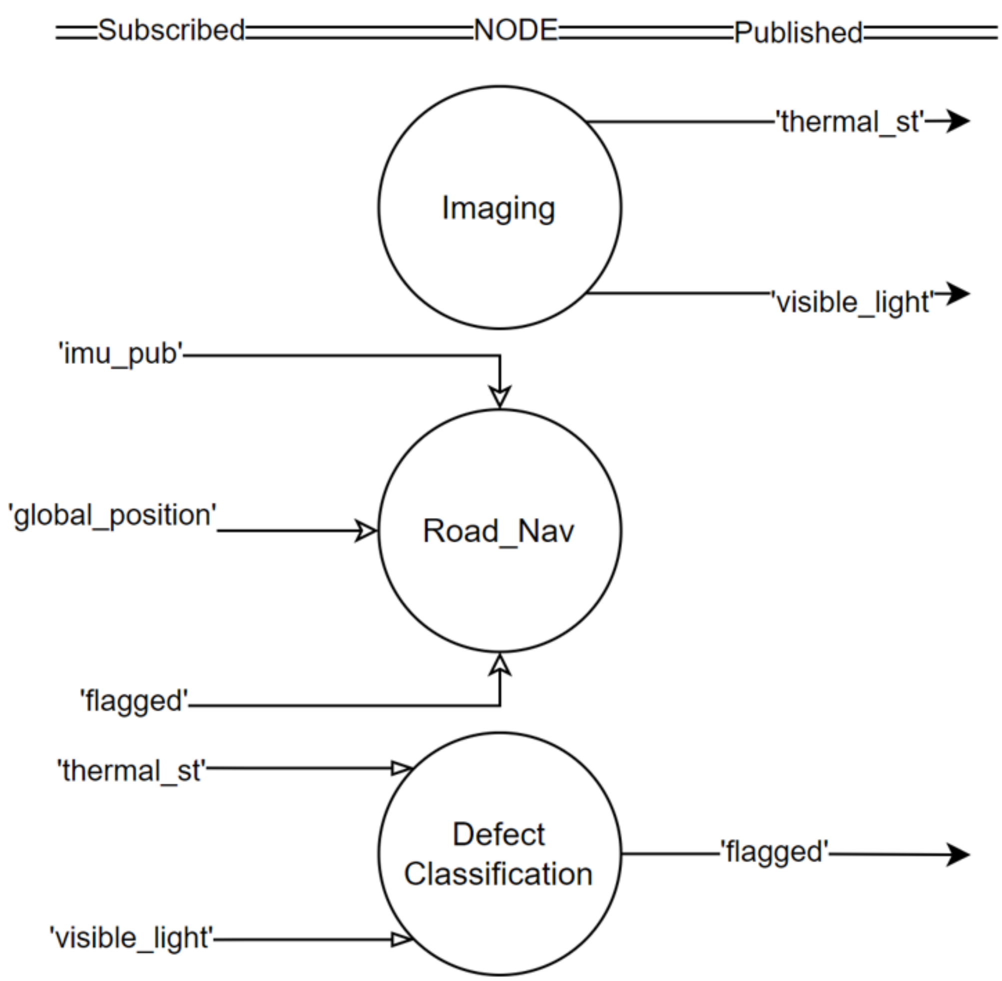

Figure 17. Block Diagram of the Three Main Nodes Developed: Imaging Node, Road Nav Node, and Defect Classification Node

Control Systems

The drone's flight controller is a square-root type proportional-integral-derivative (PID) controller, as shown in Figure 18. This allows the drone to come back to a smooth and stable flight pattern quickly and accurately. This type of controller limits the overcompensation of the system's closed loop integrations. The square-root controller allows the drone's PIDs to be easily tuned with precise calculations, minimizing the drone's chance of error and oscillation in mid-flight. With quick movements of torque, the drone's stable reaction is resilient to oscillation, moving to counteract such movement to bring it back to stationary stabilization. To produce these values, the Mission Planner used (ArduPilot) makes setting these values user-friendly while the interface shows the extended tuning branches of the flight controller, allowing users to make pre- and post-flight adjustments to the flight controller's operating system.

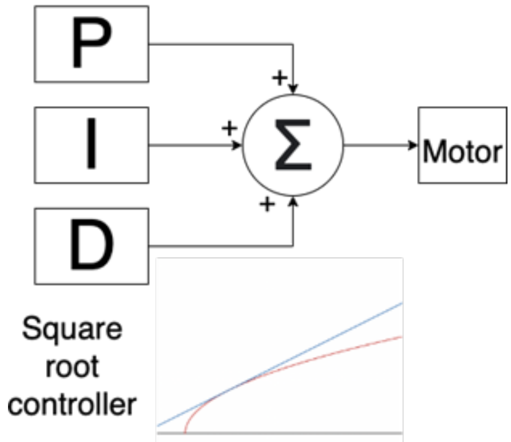

Figure 18. PID Square Root Controller

5. Machine Learing

Deep learning is a branch of machine learning in artificial intelligence. Deep learning algorithms use artificial neural networks (ANN) that replicate the functionality of a brain. It is made up of layers of artificial nodes that carry raw input data through each layer to the final output layer. These neural networks are powerful in decision-making and can learn from unstructured data. A deep convolutional neural network (DCNN) model was created in this project, as DCNN is the most common for image classification. The architecture of a DCNN algorithm implemented in this project is shown in Figures 19 and 20, for the visible light camera image input and the thermal camera image input, respectively. See the description below for additional details. The architecture for both is the same, they differ only in the input images to the algorithms.

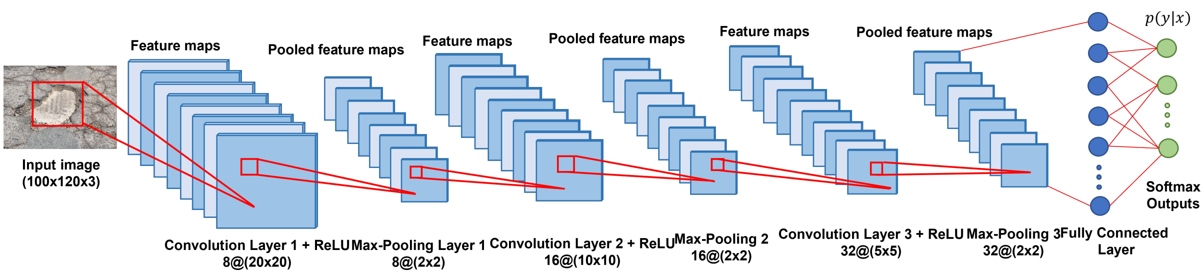

Figure 19. Convolutional Neural Network Architecture for the Visible Light Camera Image Input

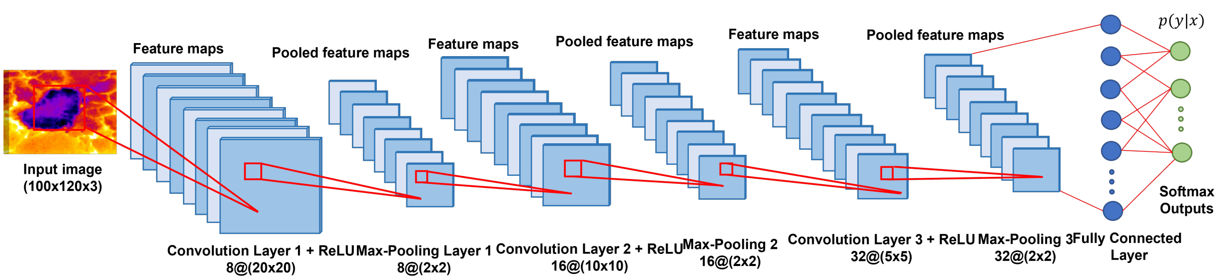

Figure 20. Convolutional Neural Network Architecture for the Thermal Camera Image Input

Sample images of data collected with the visible light camera and thermal camera that are used to train the neural network are shown in Figures 21 and 22, respectively.

Figure 21. Sample Images of Potholes and a Manhole Cover
Gathered with the Visible Light Camera

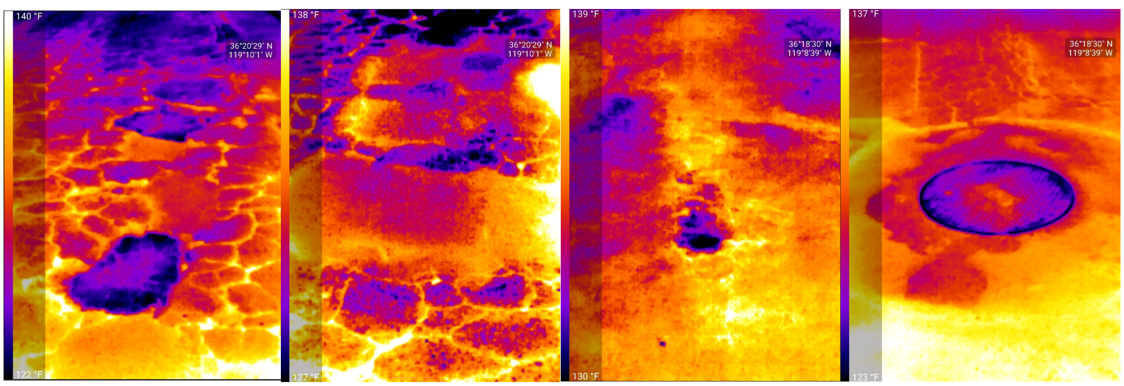

Figure 22. Sample Images of Potholes and a Manhole Cover
Gathered with the Thermal Camera

The captured visible light camera and IR camera images are first manually cropped to a 100 × 120-pixel size. The input images undergo a feature-extraction network by first being processed by the convolution layer consisting of eight convolution filters of size 20 × 20. The output from the convolution layer goes through the rectified linear unit (ReLU) function followed by the pooling layer, which employs a maximum pooling process of 2 × 2 matrices. This process is repeated several times to create the output and to train the machine with inherent features of the image. The output of the pooling layer is fed into a second convolution layer consisting of 16 convolution filters of size 10 × 10. Similarly, after passing the output through the ReLU function it undergoes the pooling layer with a maximum pooling size of 2 × 2 matrices. Finally, it passes through a third convolution layer consisting of 32 convolution filters of size 5 × 5, which is processed by the ReLU function and the pooling layer with a maximum pooling size of 2 × 2 matrices.

The classifier network consists of a fully connected layer comprised of 100 hidden nodes, which produces a Softmax output (i.e., a vector of probabilities) that in turn is used for classifying the road condition. The output layer of the DCNN represents the probability distribution containing the probabilities that each class is assigned in accordance with the input images. Using a maximum ratio combining algorithm by combining data from the two sensors, once the algorithms detect a fault it will register the coordinates where the fault is detected using the GPS signal; if no fault is detected it will continue to capture the data from the two sensors, passing it to the algorithm to perform the road/bridge inspection.

The goal of the machine learning aspect of this project is to increase the speed of data analysis while examining roadways (and, in the future, bridges). Deep neural networks allow for quick and accurate analysis of large amounts of data, though they are not infallible. For this reason, two machine learning models were adopted to work in parallel in two different ways. The first method uses an infrared camera in tandem with a visible light camera to increase the odds of a fault being detected, since the two different cameras capture different types of information. While both methods are visual inspection methods, the infrared camera detects variations of heat across the surface of the roadway. Faults were observed to have a different temperature compared to the surrounding asphalt; these are highlighted in the thermal images.

There are also two different forms of model being utilized for this project: a classification deep neural network, and a region-based convolutional neural network (RCNN). The first of the two forms, the classification of a deep neural network, is used on the drone itself. The function of the classification neural network is to quickly analyze photos and classify the photos into one of any number of groups. These models are only capable of classifying the whole image into one of several classifications, whereas the region-based convolutional neural network can localize, identify, classify, and bound objects inside the photo itself. The classification deep neural networks complete their analysis within a shorter time frame than their region-based convolutional neural network counterparts; this enables the classification deep neural networks to do a preliminary analysis of the photos to organize the data for the region-based convolutional neural network, giving images classified as "faulty roadways" a priority for analysis. The region-based convolutional neural networks were not designed to be used on the drone due to the high amount of processing power they would require relative to the Raspberry Pi's capacity. These models would reduce the overall effectiveness of the drone, as the power and time that would be used in processing photos would greatly reduce overall flight time.

Figure 23. Region-Based Convolutional Neural Network
(Visible Light) Object Detection Test

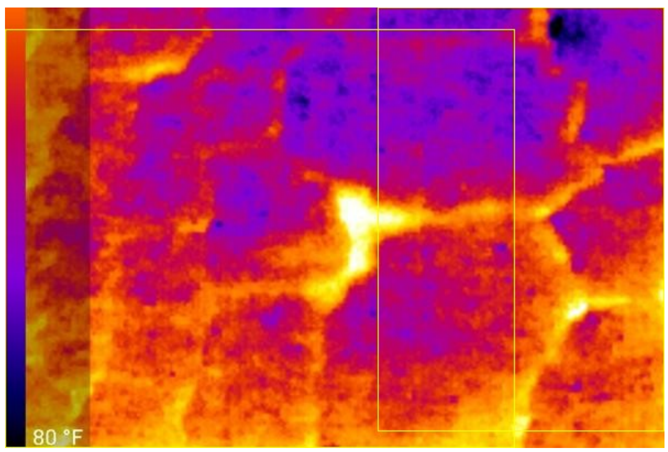

Figure 24. Region-Based Convolutional Neural Network
(Thermal) Object Detection Test

6. Drone Framework Implementation

Drone Development

During the early phases of the development, the S550 Hexacopter had trouble completing a successful flight: it fell out of the air while trying to land, due to losing communication to the motors, and other software issues, damaging the landing gear. This was caused by the drone's tuning attributions as well as the Electronic Speed Control (ESC), which desynced it from the flight controller's initial tune. The drone went through structural reconfigurations and tuning of the PIDs, allowing it to conduct more consecutive successful flights. Once the drone was stable enough to conduct missions with the sensors and the companion computer attached, data could be collected and sent to the ground station via a Secure Shell (SSH) protocol. We were then able to collect data and run it through our flagging models, allowing us to train and test how successful it was in finding discrepancies in the scanned roadway structure.

ML Development

The machine learning aspect to this project was designed to be used both with and without the drone. The smaller classification models were created to provide initial organization for the photos taken by the drone. The classification deep neural networks were designed to classify the photos into one of two groups: "faulty roadways" or "okay roadways". In addition, the design of the model included a feature in which photos classified as "faulty roadways" would be saved with a flag, allowing subsequent analysis to prioritize the photos with faults present. The Faster-RCNN models would then analyze each photo taken, starting with the photos that were flagged as faulty. These models were built using MATLAB, as MATLAB has a large number of toolboxes and accompanying documentation that is useful when implementing deep learning models. In addition, MATLAB automatically graphs training data in real time as the deep neural network is being trained. MATLAB also has several pre-trained deep neural networks that can be downloaded, modified, and used either with the MATLAB code or the Deep Network Designer. The classification of deep neural networks was modified using the Deep Network Designer. The region-based convolutional neural network was built using MATLAB code and accepts images of the same dimensions. MATLAB's Image Labeler application was used to annotate and create the MATLAB data stores required for the region-based convolutional neural network.

Software Device Integration

Using the ROS2 application programming interface (API), the imaging, road navigation and defect classification node classes were developed, inheriting members from the parent ROS2 node template class. This provided layers of abstraction, allowing for a more centralized development focus. To interface with the physical sensors, the imaging node uses two separate libraries: OpenCV and libseek-thermal, an open-source image processing library and open-source device driver library, respectively. For both the visible light and infrared sensors, the devices were opened as an OpenCV VideoCapture object, which is a class built into OpenCV to capture a sequence of frames (video) from multiple sources. Each frame processed from the VideoCapture object is an OpenCV Mat type, which is a dimensional array matrix representation of the pixel data. It is important to note that the VideoCapture objects remained open during the life of the node object instead of repeatedly opening and closing during each frame grab. This optimization saves notable amounts of computation by eliminating the need to free buffers by releasing hardware interfaces at the 9 Hz speed the infrared is limited to. This is done by opening the devices at the constructor of the class and releasing at the destructor. The node class publishers are implemented using a timer callback function supported by the chrono C++ library to achieve the desired 9 Hz data publishing rate, which translates to 9 frames per second. This function acts as a timer interrupt, executing at a certain interval. Similarly, in the case where subscriptions are made in a node, like Road Navigation, callback functions are executed any time a new topic is received, thereby synchronizing with the publisher's rate.

7. Road Navigation Using Road Edge Line Detection

To perform GPS-less navigation, we explored different algorithms to allow the drone to navigate through the streets to scan and inspect the pathways for any defects. We present the different methods we used to perform road navigation.

Multiple computer vision algorithms were used to detect and locate road edge lines in the camera frame. These algorithms were implemented using the computer vision programming library OpenCV. The first and most crucial method of detection used was Canny edge detection, which is a popular multistage edge detection algorithm that can detect changes in color gradient intensity. This algorithm uses non-maximum suppression to locate the local maxima gradient contrast pixels within a frame, and then hysteresis thresholding is used to compare these local maxima to a specified range, to determine which pixels will be finally returned as edges (OpenCV, Canny Edge Detection, 2023). The advantage of an algorithm that uses hysteresis thresholding is that it enables a more adaptive detection model to be created, since the threshold range is based on the relative edge spread of that specific frame. This algorithm detected clear changes in gradient intensity of the asphalt and the painted road edge lines.

Figure 25 shows the canny edge detection performed on a roadway; the left side image is the original and right side is the modified one.

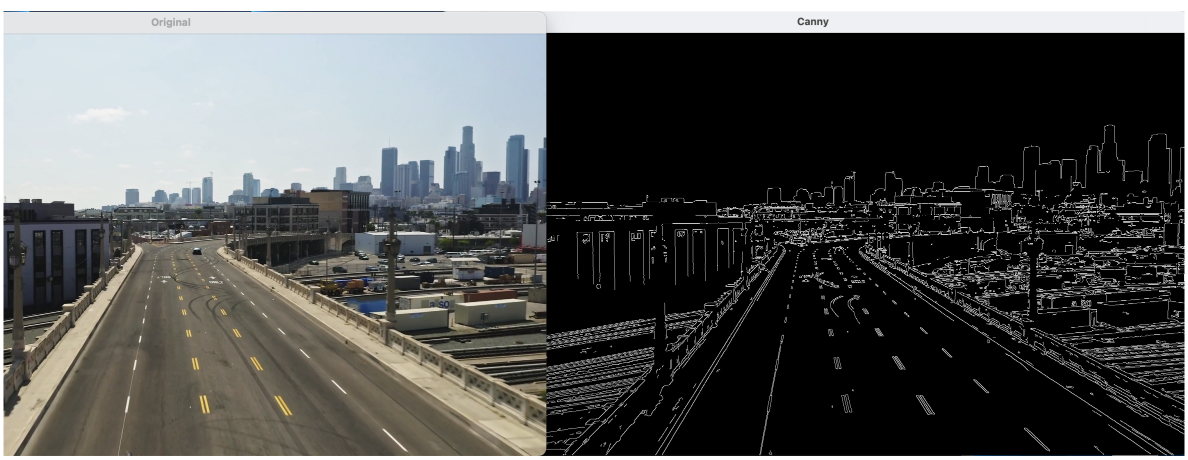

Figure 25. Canny Edge Detection Example on a Roadway Image:
Original (left) and Modified (right)

The second layer of computer vision detection used was the Hough Transform, which is a mathematical transform technique used to detect straight lines in an image frame. It does this by plotting all initially detected lines as sine curves, and defining all intersection points of neighboring sine curves to be points along the same line (OpenCV, Hough Transform, 2023). The length of the lines detected depends on the density of intersection points in the given pixel cluster, and user input thresholding is used to determine the minimum number of intersection points necessary to be detected as a line. The thresholding of this transform is fine-tuned to only detect line lengths greater than a specific high value, and clear isolation of the lines spanning the painted road edge lines in image frames was effectively achieved through extensive testing.

Figure 26 shows the Hough transform performed on a roadway; the left side image is the original and the right side is the modified one.

Figure 26. Hough Transform Performed on a Roadway Image: Original and Modified

An alternative method of detection that was utilized to detect boundary lines in image frames where edge detection was not the most effective solution was color masking. Instances in which this method proved to be more effective were when analyzing slightly blurry camera feeds or light-colored roadways. This method was used to define colors according to hue, saturation, and value (HSV) parameters and isolate those software-defined colors in the image frames. This enabled clear isolation of the painted road edge lines from the road color. After these lines were isolated, a Hough transform was then performed on the image frame to detect the visible colors as lines.

Figure 27 shows the color masking along with the Hough transform performed on a roadway; the left side image is the original, the middle image is after color masking and Hough transform, and the right side includes the green lines for navigation.

Figure 27. Color Masking and Hough Transform Performed on a Roadway Image

These computer vision methods were used to detect road edge lines in an image frame, but also to output x-axis and y-axis coordinates of the start and end points of each line detected in relation to the size of the image frame being analyzed in pixels. To determine whether the drone had passed a detected road edge line during flight, the x coordinates of these lines were compared to the x coordinate value of the vertical asymptote in the exact middle of the frame. If the x coordinates of the lines were to become greater or less than the middle x coordinate, depending on which side the line had started on, a respective left or right edge was returned. This edge information was fed into a MAVROS program (a ROS package allowing drone communication) and used to determine the attitude control of the drone. If an edge is returned by the computer vision program, the MAVROS program will send a command to the drone that will stop its motion across the returned edge. This detection control setup enables fully autonomous flight of the drone along both straight and curved roadways without the use of GPS.

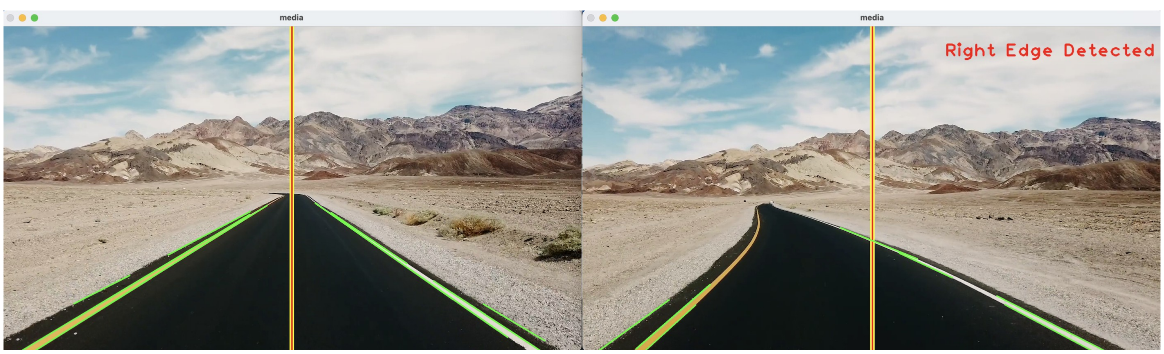

Figure 28. Detecting Road Edge Lines for Attitude Control

Figure 28 shows the process of detecting road edge lines, which is used for altitude control of the drone.

8. Prototype Experimentation

The drone was successfully built and tuned, and the auxiliary components were integrated into the drone. Thermal and optical cameras were mounted on the drone, and software was created to ensure functionality. Additionally, the machine learning aspect was successful in constructing multiple classification deep neural networks as well as the requisite region-based convolutional neural networks. These models have a high degree of accuracy whose training data set results can be seen in the machine learning experimental section below.

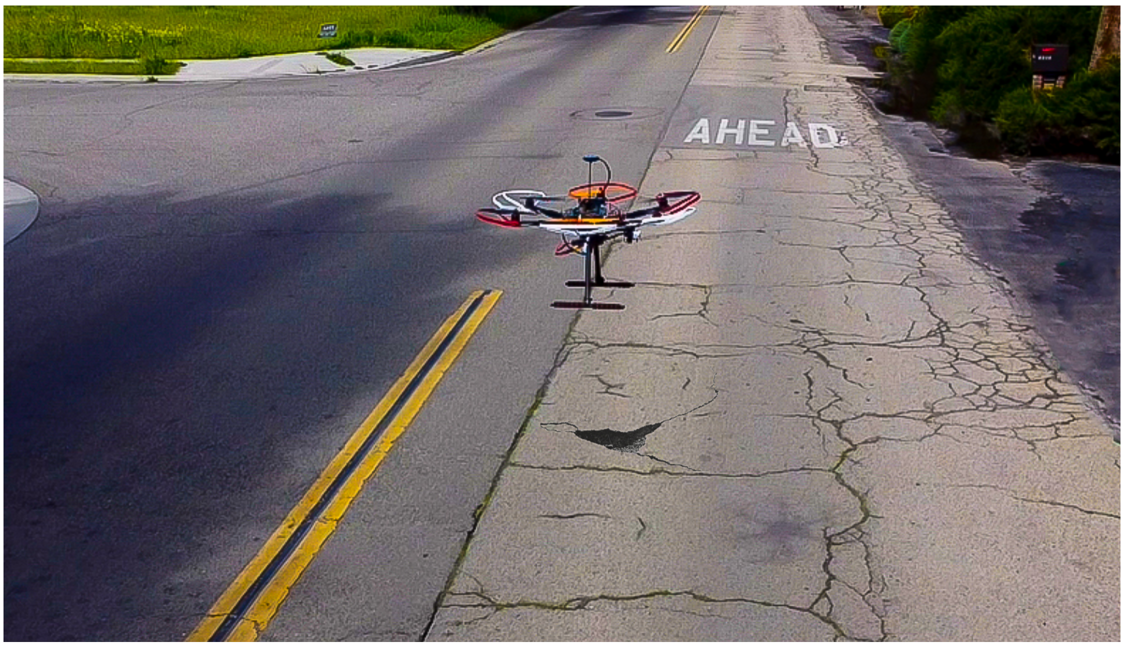

Figure 29. Experiments Conducted to Detect Faults in the Road
Using the Prototype Quadcopter Drone

All models that had been constructed for this project demonstrated a fairly high degree of accuracy. The Optical Image Deep Neural Network achieved a validation accuracy of 84.6%, and its thermal imaging counterpart achieved a validation accuracy of 95.1%. The Optical Faster-RCNN model was able to achieve a mini-batch (i.e., training) accuracy of 99.5%, whereas the Thermal Faster-RCNN model was able to achieve a mini-batch accuracy of 98.9%. Figures 30–35 below show the training results for all four models.

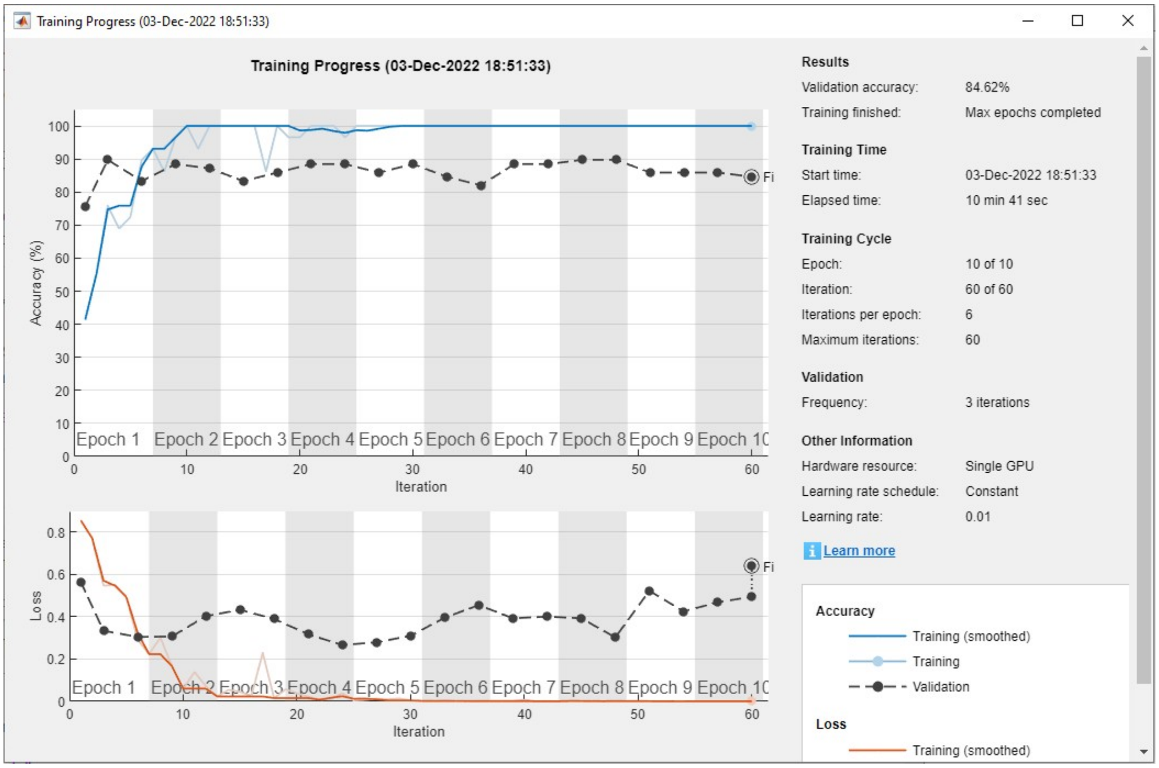

Figure 30. Training Results of the Classification Deep Neural Network for Optical Images

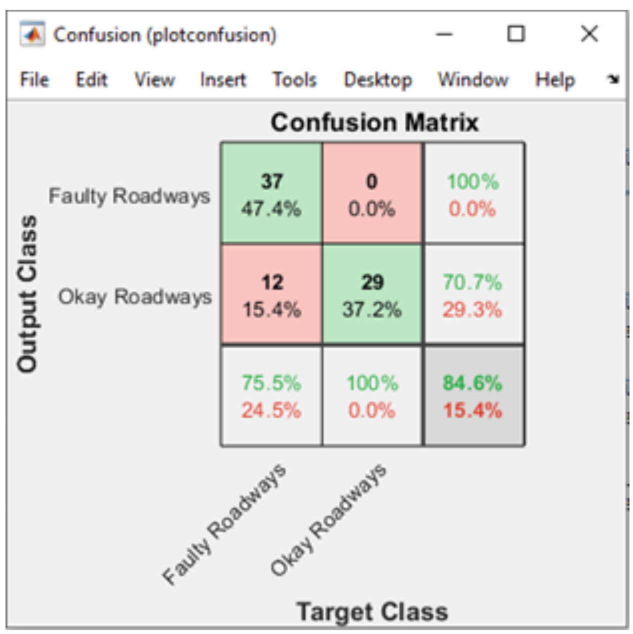

Figure 31. Confusion Matrix for the Classification Deep Neural Network for Optical Images

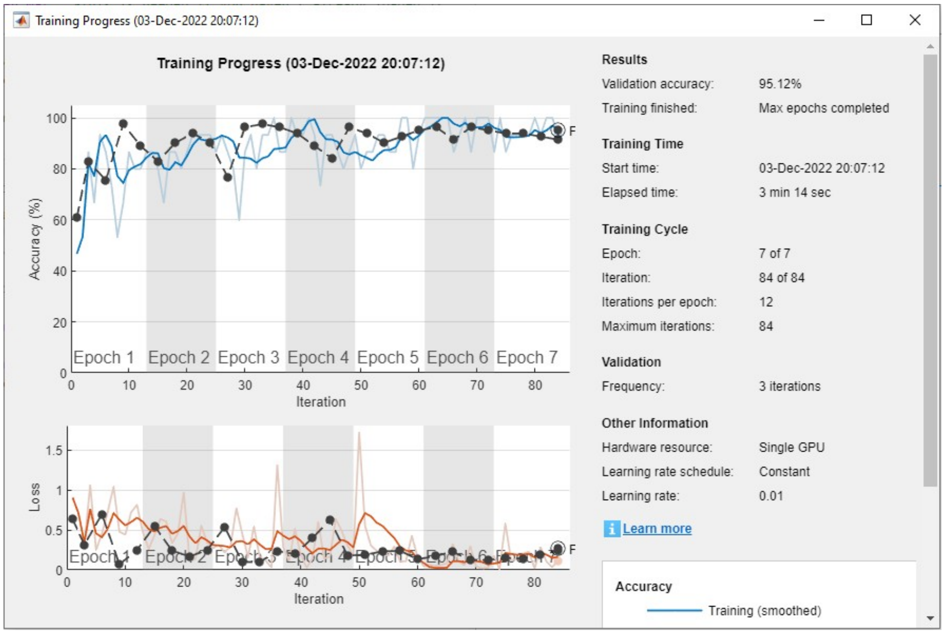

Figure 32. Training Results of the Classification Deep Neural Network for Thermal Images

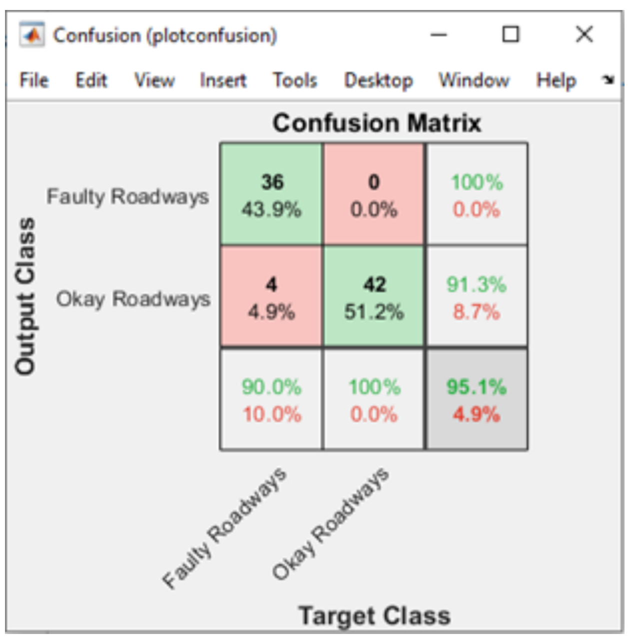

Figure 33. Confusion Matrix for the Classification Deep Neural Network for Thermal Images

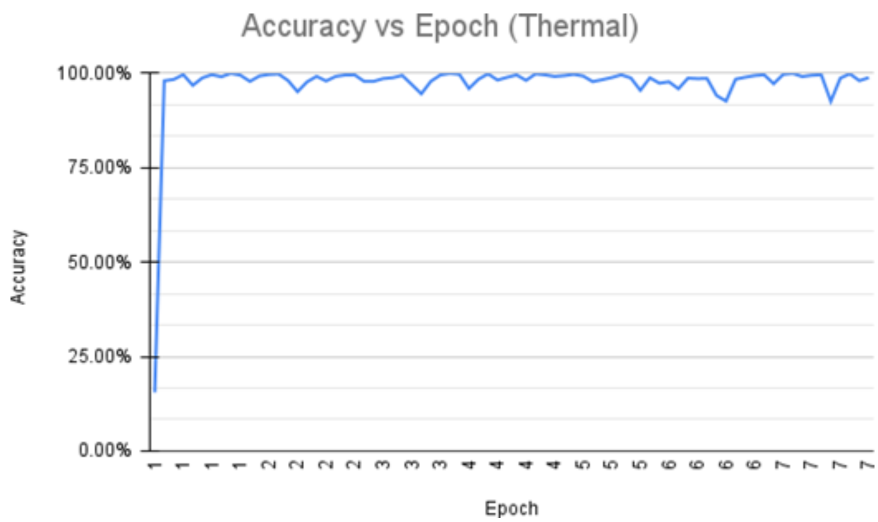

Figure 34. Mini-Batch Accuracy Plot for the Region-Based Convolutional Neural Network for Optical Images

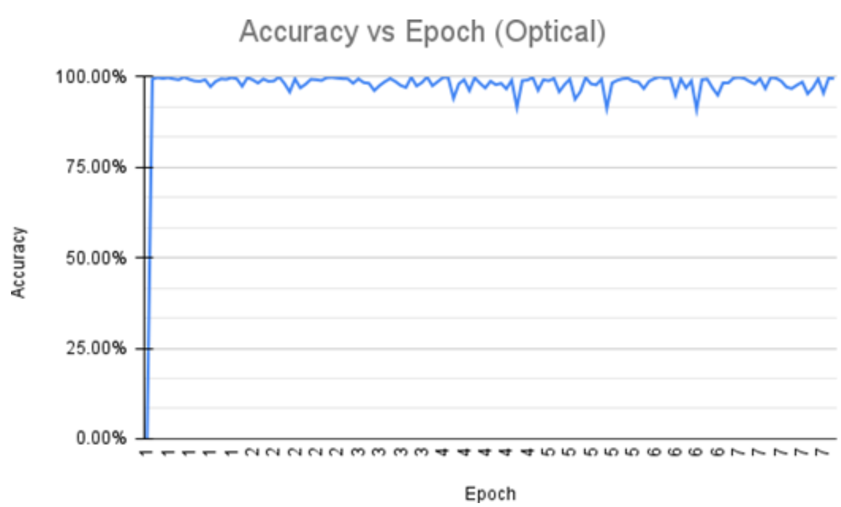

Figure 35. Mini-Batch Accuracy Plot for the Region-Based Convolutional Neural Network for Thermal Images

Machine Learning Limitations:

The greatest limitation of the machine learning aspect of this project is its limited data set size. The optical image data set only had a total of 293 photos, and the thermal image data set had a total of 282 photos. Ideally, the number of photos we would have in each data set would be around 700 photos or more per classification data set based on the research findings. Several unforeseen circumstances caused a delay in the drone's operational readiness, which, in turn, limited data collection efforts. In addition, this project only had one operational thermal camera, which meant that it could either be used to collect data or be used while integrating all the auxiliary components.

Additionally, the region-based convolutional neural network requires a large amount of time to adequately train on either the thermal or optical data sets, even while using an NVIDIA GTX 2070 graphics card.

FAA Part 107 Class Drone License:

For legal reasons, to operate the drone, one of our research team members had to acquire a Federal Aviation Administration (FAA) Part 107 Drone License. The Part 107 license demonstrates that one understands the regulations, operating requirements, and procedures for safely flying an Unmanned Aircraft (UA). If planning to fly for compensation under FAA regulations, you are required to possess a Part 107 license. For recreational use you can go through the TRUST committee to be granted without a license. In Fresno County the airport covers most of the airspace which is a Class C airspace meaning the airspace goes up to 4400 ft MSL (mean sea level). Most UA scheduled flights are done in Class E airspaces, which is everything that is not a designated airspace. To fly in this airspace, there must be a flight plan submitted to the FAA and the airport that the UA will be flown in. While flying under a Part 107, UAs cannot fly more than 400 ft AGL (above ground level) unless they are a part of inspection, which allows having a radius of 400 ft around the peak of the highest point of the infrastructure that is being inspected.

9. Conclusion

In this research project, we developed an AI-based bridge and road inspection framework that can be used to inspect bridges and roads using drones. We implemented several algorithms for GPS-less navigation of the drone using road edge line detection. Data gathering was primarily done on roads for now, using a visible light camera and a thermal IR camera. After gathering the data, we trained several Machine Learning models to automate the road inspection process by identifying sections with locations of road that need repair. Experimental results reveal we can detect defects in a road with over 95% accuracy. Additional work will be required for bridge inspection.

Abbreviations and Acronyms

AI	Artificial Intelligence
AGL	Above Ground Level
ANN	Artificial Neural Network
API	Application Programming Interface
DCNN	Deep Convolutional Neural Network
ECE	Electrical and Computer Engineering
ESC	Electronic Speed Controller
FAA	Federal Aviation Administration
FHWA	Federal Highway Act
GPR	Ground Penetrating Radar
GPS	Global Positioning System
HSV	Hue, Saturation, and Value
IMU	Inertial Measurement Unit
IR	Infrared
MSL	Mean Sea Level
MnDOT	Minnesota Department of Transportation
NBIS	National Bridge Inspection Standards
NTSB	National Transportation Safety Board
PID	Proportional-Integral-Derivative
PWM	Pulse Width Modulation
RCNN	Region-based Convolutional Neural Network

ReLU	Rectified Linear Unit
ROS	Robot Operating System
SDK	Software Development Kit
SSH	Secure Shell
UA	Unmanned Aircraft
UAS	Unmanned Aircraft System
UAV	Unmanned Aerial Vehicle
USDOT	US Department of Transportation

Bibliography

Biscarini, C., Catapano, I., Cavalagli, N., Ludeno, G., Pepe, F. A., & Ubertini, F. (2020). UAV photogrammetry, infrared thermography and GPR for enhancing structural and material degradation evaluation of the Roman masonry bridge of Ponte Lucano in Italy. NDT & E International, 115, 102287.

Cho, A. (2018), "Drones Give Inspectors a Closer Look at Bridges". Available at: https://www.enr.com/articles/46008-drones-give-inspectors-a-closer-look-at-bridges

Flyability, "Bridge Inspections: A Complete Guide", Flyability.com, 2022. [Online]. Available: https://www.flyability.com/bridge-inspections#:~:text=Bridge%20inspections%20are%20a%20crucial,they%20develop%20into%20major%20issues.

OpenCV, "Canny Edge Detection". [Online]. Available:https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html .

OpenCV, "Hough Line Transform". [Online]. Available: https://docs.opencv.org/3.4/d9/db0/tutorial_hough_lines.html.

National Transportation Safety Board (NTSB), "Collapse of the Fern Hollow Bridge", 2023, Available at: https://www.ntsb.gov/investigations/Pages/HWY22MH003.aspx

Seo, J., Duque, L., & Wacker, J. (2018). Drone-enabled bridge inspection methodology and application. Automation in Construction, 94, 112-126.

United States Department of Transportation (USDOT), "Highway Bridge Inspections". Available at: https://www.transportation.gov/testimony/highway-bridge-inspections.

United States Department of Transportation, Federal Highway Transportation (USDT-FHWT), "Better, faster methods for determining concrete bridge deck conditions". Available at: https://www.fhwa.dot.gov/goshrp2/Solutions/Bridges/R06A/Nondestructive_Testing_for_Concrete_Bridge_Decks

United States Department of Transportation, Office of the Assistant Secretary for Research and Technology (USDT-OASRT), 2019. Available at: https://www.itskrs.its.dot.gov/its/benecost.nsf/ID/85053cfd20822b66852583e0004a433e?OpenDocument=&Query=Home

Wells, J., & Lovelace, B. (2017). Unmanned aircraft system bridge inspection demonstration project phase II final report (No. MN/RC 2017-18). Minnesota Dept. of Transportation. Research Services & Library.

About the Author

Hovannes Kulhandjian, PhD

Dr. Hovannes Kulhandjian is an Associate Professor in the Department of Electrical and Computer Engineering at California State University, Fresno (Fresno State). He joined Fresno State in Fall 2015 as a tenure-track faculty member. Before that, he was an Associate Research Engineer in the Department of Electrical and Computer Engineering at Northeastern University. He received his B.S. degree in Electronics Engineering with high honors from the American University in Cairo (AUC) in 2008, and his MS and PhD degrees in Electrical Engineering from the State University of New York at Buffalo in 2010 and 2014, respectively. His current research interests are in applied machine learning, autonomous vehicle navigation, wireless communications, and networking, with applications to underwater and visible light communications and networking geared towards Intelligent Transpiration Systems (ITS).

Dr. Kulhandjian has received five research grants from Fresno State Transportation Institute (FSTI), and he also received the Claude C. Laval III Award for Commercialization of Research, Innovation and Creativity 2021 as well as the Claude C. Laval Award for Innovative Technology and Research 2020 at Fresno State. In April 2021, as a Principal Investigator (PI), he received a grant from the Department of Defense (DOD) Research and Education Program for Historically Black Colleges and Universities and Minority-Serving Institutions (HBCU/MI) Equipment/Instrumentation, to establish a “Secure Communications and Embedded Systems Laboratory at Cal-ifornia State University, Fresno".

Dr. Kulhandjian is an active member of the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE). He is a Senior Member of IEEE. He is serving as a Guest Editor for a Special Issue "Advances in Intelligent Transportation Systems (ITS)". He has served as a Guest Editor for the IEEE Access Special Section Journal, Session Co-Chair for the IEEE UComms’20 Conference, Session Chair for the ACM WUWNet’19 Conference, and Publicity Co-Chair for the IEEE BlackSeaCom Conference. He also serves as a member of the Technical Program Committee (TCP) for ACM and IEEE Conferences such as GLOBECOM 2022, WTS 2022, WD 2021, WD 2018, ICC 2018, WUWNet 2020, and WiMob2019. He is a recipient of the Outstanding Reviewer Award from ELSEVIER Ad Hoc Networks and ELSEVIER Computer Networks.

MTI FOUNDER

Hon. Norman Y. Mineta

MTI BOARD OF TRUSTEES

Directors

MTI MINETA TRANSPORTATION INSTITUTE