AN AI-DRIVEN STRATEGY FOR NON-INVASIVE FAULT DETECTION:
TECHNIQUES, APPLICATIONS, AND INNOVATIONS

by

HOON LEE

A dissertation submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY IN ELECTRICAL AND COMPUTER ENGINEERING

2024

Oakland University
Rochester, Michigan

Doctoral Advisory Committee:

Ka C. Cheok, Ph.D., Chair
Das Manohar, Ph.D.
Jun Chen, Ph.D.
László Lipták, Ph.D.

To my family,

This page is dedicated to my family for all their continuous support and dedicated encouragement.

To my father, Lee WooChul, who has always endorsed education and knowledge amongst all else. The leader of the family and provider who always made sure we had everything we needed at all times. This PhD fruition is dedicated to you.

To my mother, Ku KyungJa, who has always been the foundation of the family and taught us perseverance and dedication. The caregiver who has always been supportive and by my side at all times. Always kept me grounded and felt loved.

To my sister, Lee Jin, who always reminded me of the importance of family.

My parents, who have sacrificed everything and depleted their whole life savings to support my education during the Asian financial crisis of 1997. Even going as far as, going against government regulations and sending money for tuition to the USA.

I could not have come so far without your paramount sacrifices and utmost support.

Thank you.

ACKNOWLEDGMENTS

I want to express my heartfelt gratitude to Dr. Cheok and the committee members for their unwavering support and guidance throughout this journey. Your expertise, insight, and encouragement have been invaluable, helping me navigate the complexities of my research.

Thank you for fostering a collaborative and inspiring environment that allowed me to explore my ideas and push the boundaries of my understanding. Your commitment to excellence and passion for knowledge have left a lasting impact on my academic and personal growth.

I also appreciate the constructive feedback you provided, which challenged me to think critically and refine my work. Your mentorship has truly shaped my experience, and I am deeply grateful for the time and effort you invested in my development.

I will forever cherish our journey and grateful for teaching me the power of “So What”.

Thank you once again, for being an exceptional mentor and role model.

Hoon Lee

PREFACE

The field of technology and AI has advanced dramatically recently and now has become the main focus in our daily lives. It is no longer fascinating in movies and futuristic posters. It has fundamentally changed how we work in various industries supporting to addressing of complex challenges. My dissertation titled “AI-driven Noninvasive Fault Detection System”, explores innovative approaches and AI tools to identify and detect faults in systems across various and diverse fields of applications.

My inspiration for this research came from my interest in AI tools and finding faults in machines. As technology has advanced so has the complexity in them. There are more parts and more functions as we expect more from the machines. For this reason, we need more sophisticated and efficient fault detection systems. So, with the implementation of machine learning and data-driven methods, this research aims to contribute to the ongoing research in improving fault detection using AI tools.

Throughout my research, I had the opportunity to collaborate and receive mentorship from my advisor, Dr. Cheok, for his guidance and encouragement.

I believe my dissertation and research could be a valuable resource to researchers, practitioners, and students who are interested in pursuing the field of fault detection using AI tools. The findings and validation data collected in this research will support the advancement of academic knowledge and practical applications in the real world.

Please enjoy my findings and exploration in the field of AI-driven Noninvasive Fault Detection.

ABSTRACT

AN AI-DRIVEN STRATEGY FOR NON-INVASIVE FAULT DETECTION: TECHNIQUES, APPLICATIONS, AND INNOVATIONS

by

HOON LEE

Adviser: Ka C. Cheok, Ph.D.

This paper will provide a non-invasive fault detection solution with Artificial Intelligence (AI) techniques. The system uses already available information to collect visual, audio, and vibration data for diagnostics. Because the early signs of fault are numerous, subtle, complex, and difficult to classify and detect with mathematics and signal processing, the collected diagnostic data will be processed and analyzed for unique patterns and features to be used as input to AI tools. These features will be used to train the AI tools (Alexnet, Googlenet, Hybridnet, Yamnet, LSTM, Fuzzy Logic). The subtle characteristics are learned through training; when completed, the trained AI tools can detect them in real time. Each collected data will be classified as a fault with a soft value between [0-1]. Some faults are better detected by visual images than by vibration or audio, while others are better with audio and/or by vibration because the fault is embedded deep inside a machine. For example, an Instrumental Panel (IP) showing

engine rpm, and a speedometer complemented by engine sound and wheel vibrations can reveal non-obvious anomalies that might not be shown solely on the IP. Sound and vibration would be able to provide the early telltale signs of anomalies inside the engine and wheels. These theories are experimented with and validated on an actual vehicle and will show that a non-invasive fault detection solution is a viable solution to early fault detection.

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Continued

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES - Continued

LIST OF ABBREVIATIONS

AudFN	Audio Fault Network
CNN	Convolution Neural Network
LSTM	Long Short Term Memory
MATLAB	Matrix Laboratory
NIFDT	Non-Invasive Fault Detection Tool
NN	Neural Network
PM	Preventive Maintenance
PredM	Predictive Maintenance
RM	Reactive Maintenance
RNN	Recurrent Neural Network
VibFN	Vibration Fault Network
VisFN	Visual Fault Network
VM	Vibration Magnitude
VMR	Vibration Magnitude Ratio
YAMNet	Yet Another Mobile Network

DATA ON COMPACT DISC

Program Code

1. MATLAB VisNet (Hybrid Net) training code [Content of computer file #1]
2. MATLAB VisNet (Hybrid Net) connecting code [Content of computer file #2]
3. MATLAB AudNet training and validation code [Content of computer file #3]
4. MATLAB VibNet training and validation code [Content of computer file #4]
5. MATLAB Fuzzy Inference System code [Content of computer file #5]
6. MATLAB data handling code [Content of computer file #6]
7. MATLAB Simulink code [Content of computer file #7]
8. MATLAB NIFDT test code [Content of computer file #8]
9. Arduino C Program Code. IMU data collecting code [Content of computer file #9]
10. Raw vibrations and sound data collected [Content of computer file #10]

CHAPTER ONE

INTRODUCTION

1.1. Background of Complex Machines

Machines are complex equipment developed with many integrated parts and technology and as technology has advanced as described in journal articles, so have the complexity and features. More components and functionalities are installed as more features and requirements are added to the machines. This has increased the complexity and thus the difficulty of maintaining them as well.

We rely on these vital complex machines for our daily routines and keeping them operational has become more critical than ever. Everything from the coffee machine for morning coffee to our transportation car to take us from home to work and nuclear power plants for our daily electrical needs. Maintaining these is difficult because each machine component inside could break down at any moment and the tiniest part could cause fatal damage. Furthermore, some machines are difficult to shut down and some are sealed so modifications or opening them could be expensive and troublesome. Non-invasive diagnostics methods would not require modification to operational closed-loop systems and would be able to detect early signs of failure.

Maintaining these complex machines is difficult because they are comprised of several smaller parts and each part gives different telltale signs. It could be visual, sound, vibration, heat, or some other form of sign and this is difficult to recognize for different faults because one affects the other.

The journal articles investigated how one form of sign could give us methods to recognize faults but there is a lack of investigation on techniques for combining and blending the different telltale signs to improve the diagnostic results. A hybrid of different diagnostic algorithms could extract the essential fault features and blend them to get the best possible diagnostic.

Diagnosing complex machines requires analyzing large amounts of data from each component inside the machine. Analyzing these and extracting fault features from the data and programming to recognize them could be difficult with just math and signal processing methods. These would require automation and batch processing to process such large amounts of data in real-time efficiently. AI tools are versatile and can batch-process large amounts of data.

There are many fault detection methods for using various types of sensors. In this paper, we will demonstrate how hybrid AI techniques combine these sensors and methods to perform non-invasive diagnostics methods.

1.2. Importance of Motors in Complex Machines

Electrical machines have become an essential part of everyone's lives, and it is no doubt that humans will become more dependent on them. We are highly dependent on electrical machines for transportation such as electric cars, electric trucks, and in the near foreseeable future, electric boats, and electric planes. As electrical machines become more essential, the reliability of electric motors inside has become important more than ever. The heart of any electrical machine is the electric motor inside as it is the complex power source that moves, drives, and generates energy. We will discuss the essential reasons for depending more on electrical machines.

The increase in electrical machine usage came as the government and society increasingly attempted to achieve 0% emission and became more and more aware of the necessity of having green mentalities towards machines and creating fewer carbon footprints. They targeted the most heavily used and visible carbon machines, the everyday transportation personal vehicles, and cars. As the motor companies start to produce more electrical cars like Tesla due to having more financial incentives provided by the federal government, we are more dependent on electricity to charge the electrical car batteries. But more people are realizing that the electricity that drives the cars is provided by the coal factories, which does not help elevate the greenhouse effect issue as much as it was advertised. So now there is research on the way to reduce the amount of carbon from the plants as well. CO2 taxation among others is expected to accelerate the progress of zero-emission power plants. The concept of zero- and near-zero-emission power plants can be realized by combining advanced power cycles and carbon capture technologies.

With government and public incentives for reducing air pollution, electrical machines are the future, and combustion engines are becoming the things of the past. As we are more heavily dependent on electrical machines for our daily needs it is essential to have a reliable machine with foolproof electric motors. The heart of electrical machines is the electric motors. It is the main functional component inside the machine that moves and generates power. The electric motor must be non-fault and foolproof as it is a critical component inside the electrical machine in several critical application areas that we will discuss below.

You could imagine what it would be like being in the sky inside an electrical plane and one of the motors malfunctions abruptly. Or a space station with a downed electric motor and no tools or spare parts. Or fatal accidents in heavy machines in the manufacturing and power plants that might even lead to the death of factory workers due to sudden motor failures and unscheduled maintenance activities on machines.

Not only in industrial settings, electrical machine usage inside our homes and living rooms has become a daily routine as well. There are various kinds of electrical machines doing mundane daily chores and supporting our lives, we call them robots. Starting with vacuum robots such as “Roomba” that we have seen in TV commercials to a nursing robot that takes care of the sick and elderly are just a few of the home robots available in our lives today. These nursing support robots would be helpful in pandemic situations like COVID-19 that we are experiencing where human-to-human interactions are considered deadly.

As these robots are very complex and require complicated movements and control, we need various kinds of interactive complex electric parts to meet those requirements. As these robots interact with live humans and have live interactions, the multilateral and multi-axis movements require precise motion and control and thus, need highly calibrated and complex electric motors. And as motors become complex, there are more complicated intricated components that could fail. The complex system would require a long duration of diagnostics and fault analysis to find issues and faults, if plausible. Because all intricated parts are combined in a small area, removing a part out of the machine would be a hassle.

I have explained the origin and future of various kinds of Electrical Machines and diverse types of Motors, their importance in our everyday lives and how their dependability and reliability are critical. Now we will review various kinds of Electric Motors used in Electrical Machines and known faults and countermeasures we have implemented.

1.3. Types of Faults and Maintenance Methods

There are three known types of Fault Detection and Management Methods used.

The first method is Reactive Maintenance (RM). This is a reactive fault detection and maintenance method that would wait until the machine breaks down and faults are then replaced. It is the most simple and inefficient method that comes with high replacement costs.

The second method is Preventive Maintenance (PM). This time interval method is used to replace a machine promptly before it breaks down. It is better than the first reactive method RM but still considered inefficient because we could replace still useful and operational machines.

The last method is Predictive Maintenance (PreM). This is the most desired maintenance method that could predict when the machine is likely to fail with high probability. This method would require monitoring of the signals and detecting subtle and inconspicuous tell-tale signs of faults from the machine. Careful calibration and a large amount of data are collected and monitored by machines and experts to determine the optimal replacement timing. The essential activity for PreM is early detection. It is important to recognize the subtle traits and signs of the faults that are difficult to detect.

1.3.1. Types of Faults in Motors

As previously described, I found 20 different types of electric motors implemented and used in the world of electrical motors. There are 9 different types of electric motors powered by AC signals and 11 different types of DC power. Each motor is structured differently and/or controlled using various diverse methods and algorithms. So, there are differences in hardware structure and materials used in motor motion. Due to these differences, there are also different ways a motor could break and fail. Thus, maintenance of different types of Electric motors is difficult and strenuous to repair, especially if the fault is hard to detect and breakage is sudden. Even if the sensor information can be tracked, it is close to impossible for humans to monitor all the different motor signals at once and values to predict different faults for each type of electric motor.

Below are summarized Reasons, Descriptions, and Possible known solutions to the Motor faults.

Table 1. Motor Faults. Reason and Solution

REASON	DESCRIPTIONS	POSSIBLY SOLUTION
POWER ISSUES/ UNSTABLE POWER	A sudden unstable power supply to the motor could cause issues. For example, supplying low voltage could cause excessively high current to maintain its speed and/or torque, which could damage the motor.	Have a preventive system or method to constantly monitor the current supplied to the motor and prevent overcurrent.
HIGH TEMPERATURES/ OVERHEATING	High temperature in the motor and overheating could be caused by unstable power and ambient factors, such as high friction or blockage in the cooling system.	Have a preventive system or method to constantly monitor the power supplied to the motor and temperatures in the motor and surroundings.
VIBRATION	Vibration is normal to an extent to a moving motor, but excessive vibration could be caused by misalignment, part corrosion, and loose parts such as bearings. If not attended to, this will eventually stop motor operation.	Have a preventive system or method to constantly monitor the factors related to motor vibration such as vibration values, motor speed, torque, temperatures, and power supply to determine if there is any issue with worn loose bearings or other related causes.

Each part plays a critical role in the operation of the motor and one part being faulty will decrease motor operation efficiency and eventually halt operation. Below is the motor part that fails and its known reason that I have found and summarized.

Table 2. Motor Part Fault and Reasons

PART	KNOWN REASON
SHAFT	Imbalance, Misalignment, Wear
ROTOR	Imbalance, Rotor Bar Faults, Loose Rotor, Eccentricity
STATOR	Stator Looseness, Incorrect Air Gap, Winding Fault
MOTOR BEARING HOUSING	Mechanical Looseness, Misalignment
ROLLING ELEMENT BEARINGS	Cage, Ball/Roller, Outer and Inner Race Defects Looseness within the bearing
MOTOR FEET	Structural Looseness/Soft Foot
BASE-PLATE/FOUNDATIONS	Structural Looseness, Twisted Supports
MOTOR INSULATION	Insulation breakdown, Heat due to friction, corrosion. This will lead to low resistance and end with a short circuit to the ground or another conductor.

1.3.2. Known Methods Used for Motor Fault Maintenance

We have stated various kinds of Electric Motors and the difficulty in maintaining such a large number of various types. We will show what methods the public is using to keep the Electric Motors running and why.

Machine maintenance is a big, complex, regular process. In addition, the spare parts required for this must be kept in stock until a machine fails. To avoid a production breakdown in the event of an unexpected failure, more and more manufacturers rely on predictive maintenance for their machines. This enables more precise planning of necessary maintenance and repair work, as well as a precise ordering of the spare parts required for this. A large amount of past, as well as current information, is required to create such a predictive forecast about machines. It is not just about trying to save money by using the machine until failure but also possible fatal incidents due to sudden machine malfunction in highly industrial settings.

Large Industrial machine maintenance methods can be generally divided into three categories. Reactive Maintenance would be to fix the machine on the spot as it is no longer functioning. Preventive Maintenance would be to set an effective estimated date to replace the machine before the machine breaks down. Lastly, Predictive Maintenance is collecting data from sensors in and around the machine to calculate and scientifically predict how much life is left in the machine and what part inside is showing signs of wear.

1.3.3. Difficulty with PreM implementation

The difficulty in implementing PreM is due to managing a large amount of data and analyzing them to develop a proper regression model. Compared to the two previous methods of replacing and managing damaged equipment as it occurs or maintenance of equipment or replacement at a decided interval, PreM requires more resources and complexity. PreM requires the system to accurately predict the time when maintenance is needed. This can be predicted by analyzing the previous instances and monitoring the changes as they occur within the system.

Another reason for implementation difficulties with PreM is due to lack of data. As the prediction model and monitoring are performed on collected data, when there is a lack of data to use to train models and learn from the instances, it is difficult to have a trained AI model that could predict when the equipment would require to be maintained or replaced.

Overall, it is difficult because most sensors trigger when a part fails, we need to look at it from an overall perspective to monitor the degradation of the system.

1.4. Early Diagnostic Approaches

There are many ways to diagnose issues in the early stages using machines and methods. There are annual checks and tests done at periodic times to monitor conditions.

1.4.1. Routine Maintenance and Calibration

These are routine checkups on machines to see if there are any abnormal behaviors to the equipment. For example, abnormal vibration or noise that was not detected before could indicate new issues. When the issues are found in the early stage, calibration on the equipment is performed to prevent further damage. By performing the needed calibration, could realign the causal part and possibly recover from the problem.

1.4.2. Mechanical Diagnostics

The vibration analysis and diagnosis are performed with vibration sensors and analyzers. The tools can detect equipment mount imbalance, part misalignments, and wear and tear of rotation equipment such as motors, and gears. This can be used to predict early faults by predicting the remaining life of the equipment.

The analysis of temperature in equipment is monitored for early fault signs. Tools such as thermistors, infrared cameras, and thermographic sensors are used to detect abnormal high-temperature spots on equipment. These could indicate cooling issues, abnormal heat patterns, electrical faults, and friction problems.

Analyzing the lubricant used can help determine issues at the early stages. The oil can be analyzed with analyzer lab equipment. The analysis can show contaminants, wear particles, and irregular chemicals. This could indicate possible issues and insights into the equipment and help prevent near-failures.

1.4.3. Electromechanical Systems

Using a power meter and multimeter to measure equipment voltage, current and resistance can detect any abnormalities in the equipment. The multimeter could be connected to wires and electrical points to check proper electrical values during operation. More detailed information on the voltages and currents can be viewed with the oscilloscope. The oscilloscope allows the view of signals over time and could help diagnose issues with signal integrity anomalies with waveforms and problems related to frequency.

The tracing tools could be used to trace circuits and test for faults. This tool could be used to detect open circuits, shorts, and problems with grounding.

1.5. Transition to Electronic and Computerized Diagnostics

Advancements in technology and diagnostic tools helped the transition to electronic and computerized diagnostics possible. The diagnostics involve digital sensors and devices to collect real-time data from remote devices and equipment. Various sensor data could be collected such as temperature, vibration, voltages, current, and other various measurable data. Remote monitoring in real-time makes it possible to early detect and provide diagnostics of anomalies.

1.5.1. Digital Diagnostics

For digital diagnostics, we use tools such as a multimeter to measure signal voltages and currents. If we need to measure the signals more in-depth, then we would use a digital analyzer or oscilloscope. This would give us more details of the signal in the time frame. This would be important in developing and implementing signal processing methods into the system. Other digital diagnostics would involve digital sensors such as a camera, microphone, IMU, thermistor, barometer, digital gauges and so, on to measure and provide digitized output for analysis.

1.5.2. Computerized Diagnostic Systems

Computerized diagnostic systems are systems that use digital sensors and compute a decision on the diagnosis. This is used in Industrial and other settings such as Healthcare, Infrastructure, Automotive, and Energy sectors. The Industrial consists of heavy, complex machines and equipment to handle large containers and need to know if there is any issue with the machines before critical failures. For Healthcare, machines, and equipment are used to heal human beings. A diagnostic system would provide the necessary information about the tools and equipment used and prevent any issues while curing humans. For Infrastructure, this sector builds large buildings and also maintains them. The diagnostic system would monitor and process data and provide the necessary information about the structural integrity and health of the buildings and bridges.

This would be critical before there are any issues with the structure. For Automotive, the diagnostic system monitors the components inside the vehicle and informs the driver if it senses any issue by lighting up a signal on the instrument panel. This would provide the information needed for the driver and the mechanics to find the cause of the problem before fetal accidents on highways. Lastly, in the Energy sector, the diagnostic systems would collect data and monitor the electrical grid performance. The installed smart meters and sensors provide the power usage, demand, and possible issues. This would provide the necessary diagnostic information to prevent power outages, equipment failures, and abnormal usage and thus optimize energy distribution and improve grid reliability.

1.6. Importance of Non-Invasive Diagnostics

Non-invasive diagnostics is important because it offers several merits compared to invasive diagnostic methods. It offers almost no disruption to ongoing operations or systems because it does not require any modifications or access to inner parts. It is processed while the system operates in its original state. This would be important in power nuclear plants where it is difficult to shut down to install a diagnostic system.

Being able to monitor it in overall view from outside the system. Instead of monitoring signals inside the machine, the non-invasive method allows real-time monitoring of the overall system. This would provide the necessary information such as degradation, and overall conditions in a dynamic system. Also because of the ease of installation compared to invasive diagnostic, we could implement this in various systems and be able to monitor, control, and analyze remotely. For example, in a fleet of equipment and machines.

1.7. Advantages of Non-Invasive Methods

There are several advantages of non-invasive methods compared to invasive methods. Reduced risk, better safety, cost efficiency, versatility, and ease of application in different fields.

1.7.1. Reduced Risk and Better Safety

The non-invasive methods do not require any modification to the operational system so there are reduced risk of issues that come from exposure to hazardous conditions and contamination. For example, a non-invasive method would not require a technician to open and modify mechanical systems.

1.7.2. Cost Efficient

Because there are fewer actions needed to install the non-invasive system in the operational system, there are fewer costs and resources involved. Also having this system would require less costly repairs and unforeseen damages due to sudden fatal accidents.

1.7.3. Versatility

This non-invasive system is installed externally to the operational system and does not have a rigid physical feature so this could be installed in various wide range of sectors. This would include healthcare with imaging, the industrial sector with vibration analysis, and infrastructure with monitoring structural health.

1.7.4. Ease of Application

No modification or opening of the operating system is necessary so it is easy to install on the system to monitor and diagnose system issues. This ease of application provides implementing onto a large number of systems. For example, fleets of vehicles, buses, ships, etc, and monitor and control them remotely to prevent unforeseen issues and optimize the operations.

1.8. Applications Beyond Automotive

The non-invasive diagnostic system could be applied to several sectors. Diagnostic is well known in the automotive industry with technology that informs drivers and mechanics with instrument panel warning lights and codes. Other sectors include for example as follows:

1.8.1. Industrial Equipment

In Industrial Equipment, there are complex and integrated parts and gears combined to do various tasks. It would be resource-intensive to open each piece of equipment to install a diagnostic functionality. This is especially true since downtime in the automation plant is highly expensive. Also, it is highly possible to have unforeseen fatal accidents due to humans working on the machine in an unscheduled maintenance of equipment failure.

1.8.2. Consumer Electronics

Consumer electronics would not need to be modified so that there would be less hassle for the customer. This diagnostic function would be added as an external piece that would monitor the noise, or vibration and inform the customer of possible wear of the system. This would allow the customer and the manufacturer to better analyze the issue and resolve them in the early stages.

1.8.3. Healthcare

Application of non-invasive diagnostic systems to Healthcare would improve the care to the humans. This system would provide diagnostic analysis of the X-rays to humans and detect irregularities and features that humans might miss with large quantities of data. The data and images could be too subtle that it could be easily misdiagnosed seen with human eye.

1.9. Research Problem

The research problem in non-invasive diagnostic methods is noise in data, which leads to false positives and there is a lack of and incomplete data to analyze and compute algorithms. The data is crucial to non-invasive diagnostic methods because of the feature and pattern training and recognition involved in using it. Without proper reliable data, developing an operational diagnostic system is a problem.

1.10. Challenges with Traditional Methods

Traditional methods of diagnosis are widely used and provide merits to finding issues and solutions to problems. But there are challenges with traditional diagnostic methods such as follows:

1.10.1. Limitations in Accuracy and Efficiency

Because of a lack of complexity and defined methodology, subtle and hard-to-detect faults could be missed or misdiagnosed as false positives or false negatives. This is especially true for early-stage faults due to their resolution limits and sensitivity to the traditional methods.

1.10.2. Invasive and High Cost

Traditional methods mainly involve invasive procedures to opening up machines and equipment to collect data if no port is available. This requires physical intervention and disrupts normal operations or modifications to the system being tested. Such as disassembling machinery which can lead to operational halt and potential damage. The downtime in an assembly plant would lead to high costs having several assembly lines non-operational.

1.10.3. Limited Scope and Flexibility

Traditional methods have limited scope to specific issues and types of faults. It is not designed to diagnose different scenarios and evolving issues at the early stages of faults. So these methods are rigid and not easily adaptable to new technologies and changes which would limit their use in dynamic environments.

1.10.4. Difficulty with PreM Implementation

Modifying closed-loop systems and machines that are sealed and operational could be difficult. For example, nuclear reactor systems require difficult activities to modify a machine to add new PreM features. We should consider implementing this in a non-invasive approach that would not require any kind of modification to the operating system.

It is difficult to detect early signs of faults because faults are exposed from the combination of various factors. For example, visually a machine could look fine, but it could be making unknown noises. Or it might be making a noise, but you could feel emulating heat from the machine. The signs are conspicuous and subtle and have different variations, features, and characteristics that are difficult to recognize. This system would require collecting and processing data from combinations of sensors to better detect early faults.

To be able to continuously record and process large amounts of data and recognize subtle fault features is difficult with only math and signal processing methods. Such a large amount of data would require smart batch processing tools and methods implemented with signal processing techniques. This would require AI tools.

1.11. Opportunity with AI-Enhanced Non-Invasive Methods

There are significant opportunities with AI-enhanced non-invasive methods that would address many issues with traditional methods and provide the necessary functionality to improve diagnostic solutions. These methods would improve the accuracy, efficiency, and reduce false positives and negatives, and improve the reliability of the diagnostic solutions. For example, improved precision and accuracy, real-time monitoring with a predictive maintenance model, enhanced machinery insight with data analysis, and improved decision-making would be the advantages of having AI-enhanced non-invasive methods compared to traditional diagnostic methods.

AI-based events classification methods described here use example as fault detection, to detect anomalies as the use case. AI tools can process large amounts of data and external sensors that monitor complex and comprehensive overall system output can monitor the wear and tear of the system as a whole which could predict and provide the necessary tools for predictive maintenance.

CHAPTER TWO

LITERATURE REVIEW

2.1. Potential for Real-Time Fault Detection and Predictive Maintenance

There is potential for real-time fault detection and predictive maintenance in all sectors. This would provide an optimized diagnostic and monitoring system for systems and hardware equipment. The AI-enhanced non-invasive methods would provide significant potential.

2.1.1. Integration of AI with Non-Invasive Sensors

The integration of AI with non-invasive sensors would provide the power of AI with data-collecting sensors without modifying or opening an operational system and possibly causing damage and operation downtime. Integrating the AI would provide the functionality to analyze all the data from the sensor and collect the features and patterns. This would allow the AI to learn from the data and predict any faults with the collected data's features.

2.1.2. Case Examples

Case examples of the potential for real-time fault detection and predictive maintenance are from Industrial Turbomachinery and Aircraft Health Monitoring.

Industrial Turbomachinery at General Electric (GE): This case study is from GE on its turbomachinery and how the AI fault detection system would provide benefits and solve its problems. The GE turbomachinery is used to provide power generation and there are unexpected downtime to repair mechanical issues and failures that are very costly. The problem with traditional methods involved reactive methods which with periodic inspections resulted in unplanned outages and maintenance costs.

However, after implementing an AI detection system and using learning algorithms to monitor and analyze data from the sensors on the turbines, there was reduced downtime. The AI system monitors the signals for learned features and predicts any possible issues. This is done in real-time and saves future high costs and reduced resources. With AI detection system, this improved the overall system and reliability of the turbomachinery turbine.

Aircraft Health Monitoring at Airbus: This is another case study that showed the benefits of implementing AI-powered detection systems. The case study showed that traditional fault detection methods of interval inspections could not meet the high demands of reliability of aircraft fleets. So Airbus implemented an AI-driven fault detection system to monitor their aircraft. The data are collected and analyzed in real-time showing its engine performance, avionics, and fleet's health. This has provided the necessary information to predict potential faults and maintenance needs. The result of enhanced safety, increased efficiency, and reduced maintenance cost from this case proved the benefits of the implementation of AI-driven fault detection and monitoring systems.

2.2. Case Studies Illustrating Common Problems

To demonstrate events in real life with common problems with traditional diagnostic methods, below cases studies are introduced as follows.

2.2.1. Industrial Machinery

The industrial machinery is maintained by monitoring the mechanical vibrations. This is to check the status of the rotating machinery for example, motors and pumps. The monitoring system can detect and identify issues such as imbalances and misalignments due to improper installation but noticed several limitations. Such as the false positive results from the vibration sensors. The warnings and results from the vibration sensors stating errors ended up not being true errors. For example, the vibration sensor alarmed that there are severe issues when the excessive vibration, in the beginning, is normal to start up the machinery.

Another issue is due to the limited scope of the vibration sensor. The sensor sensitivity was not enough to warn about the early wear and tear and ended up costing repairs and unexpected breakdowns.

Lastly, because installing the sensors required opening and accessing inside the machine, this costed resources, especially when each machine needed modifications.

2.2.2. Automotive Maintenance

Automotive maintenance is performed by monitoring vehicle parts' status and reporting them through LED lights on the instrument panel and internal diagnostic system. This is a valuable diagnostic system for drivers and mechanics but there are limitations. The On-Board Diagnostic (OBD) provides the status of the vehicle parts but it is limited to engine-related issues and may not detect problems in transmission and suspension or combination of the parts. This would prevent the ability to predict future issues and prevent failures shortly.

The Diagnostic Trouble Codes (DTCs) that are available to check the faults in the vehicle could be ambiguous and not helpful. This would require further investigation and the code not being useful to resolve the vehicle issue. This would result in the mechanic investigating all the parts indicated in the code and not finding the root cause of the problem. For example, diagnosing non-engine-related issues and constantly chasing the diagnostic codes would lead to added costs and resources.

2.3. Objectives

The objectives of this dissertation are to provide research and documentation of how AI-driven fault detection systems are beneficial in different sectors. Case studies and research papers are shared and referenced in this dissertation. Actual programs written in scripts of various programming languages will be shared as well. The program will be integrated with different AI tools and trained with various layers and models. The performance and limitations will be analyzed and discussed. This will be tested in real-time on a sedan vehicle. The test case would be vehicle starting issue fault detection using the AI-driven program. The program will analyze the data and provide the predicted fault in real time. As the starting issue shows very similar symptoms in visual, sound, and vibration, the AI-driven fault detection system would need to be able to properly predict the real fault with uncertainty.

2.4. Primary Goals

The primary goal is to analyze the necessary functionality to predict and detect faults. The faults will need to be analyzed for features and patterns and collect data from various angles using different sensors. The sensors and the system are non-invasive and will attain the data from outside the operating system. There will be no disruption to the operational system at any time and will not require any access to internal parts, such as plugs or connections. This analysis and detection is performed as solely as an observer to the operational system. By analyzing multilayer sensors and their data, the detection system would provide the highest probability of fault in real time.

2.5. Developing a Non-Invasive Diagnostic System

Developing a non-invasive diagnostic system would require hardware and software to run the system. The hardware would consist of sensors and computing resources such as processing units and sensors collecting surrounding features and data.

2.5.1. System Design

Laptop is used to develop necessary scripts and programs installed with MATLAB and its tools. Arduino development interface and tools are also installed to collect data from IoT (Internet Of Things) hardware that is not available with laptop peripherals.

2.5.2. Integration of AI Tools

Integration of AI tools would include Convolution Neural Network (CNN) to analyze and detect features and patterns in visual data. This is visual frame data collected with a camera that captures movement frames.

Audio data is captured and analyzed with CNN. This network will analyze the audio data in time and spectrum and recognize the features.

The sequence of the vibration data is analyzed by Long Stort Term Memory (LSTM). The sequences are analyzed for features and will be able to detect any patterns of a fault.

Lastly, all the collected data are combined and computed with Fuzzy Inference System (FIS) and provide the most likely fault. All these AI tools are integrated.

2.5.3. Proposed System

The proposed system is a Non-Invasive Fault Detection system with AI Tools (NIFDT) that consists of a hybrid of AI tools and decision-making FIS. The program is developed and run in MATLAB and IMU controlled program is scripted in Arduino IDE. The NIFDIA consists of sensors data recording devices and a laptop.

The NIFDT consists of characteristics that it is non-invasive does not require any direction connection to the operational system has multiple sensors and runs with a hybrid of AI tools. It is non-invasive and requires no modification to the operational system and no stoppage.

This proposed solution is non-invasive, meaning no modification to the operating system is necessary. This external system with trained AI tools would record raw data in the surroundings and detect faults.

Multiple sensors can be connected to NIFDT. Each sensor will provide its unique data to help determine the fault. Multiple sensors are included in the system to provide accuracy and reduce false positives. Monitoring from different perspectives would reduce the probability of producing false positives and negatives.

AI tools: NIFDT has a hybrid of AI tools that are integrated to provide accurate analysis of the data captured from the multiple sensors. AI tools are chosen specifically for their traits and abilities. The CNN will analyze the visual and audio data and format them accordingly to extract the necessary features.

2.6. Evaluating System Performance

NIFDT's performance will be evaluated and measured for possible comparison to other methods and improvements. Its limitations and shortcomings will be discussed with performance data.

2.6.1. Comparison with Traditional Methods

The performance of NIFDT will be compared with traditional methods. For example, we would compare the fault detection system on the car compared to using NIFDT in real time and compare the merits and shortcomings of both methods.

2.6.2. Performance Testing

Performance testing is performed with resulting accuracy. We will determine how many resulting outputs from the system are accurate and which are false positive or negative. Other performance would be for time latency and how long it takes to assess the diagnostic data and produce a viable predicted fault.

2.7. Specific Aims

The specific aim of this research is to assess the capabilities of fault detection with sensors and AI tools based on being non-invasive. The hypothesis and theoretical methods are tested in real life and analyzed on their performance and limitations. The future actions would include resolving or reducing the limitations and providing functionalities for more applications to the real world.

2.8. Designing System Architecture

NIFDT system architecture is designed with both hardware and software. The hardware and software are integrated to collect data in the surroundings and provide reliable predicted faults. The design consists of sensors and controlling software that is specifically designed for data collection and processing.

2.8.1. Hardware and Software Components

The hardware consists of different sensors that collect data and properties of the surroundings. This would involve sensing the changes and properly detecting the values. The software is in MATLAB program and scripts are written in C for Arduino microprocessor. The hardware is connected through cables from Arduino hardware pins to IMU and possibly other sensors to communicate and transfer data. The Arduino is connected to a laptop personal computer through a USB-A cable for sharing data and control of the script program in Arduino.

2.8.2. System Integration

The system integration of the software side involves the development and compilation of the source code written in MATLAB and Arduino IDE. The training and validation of the AI tools and models are compiled and integrated within MATLAB. The AI models and FIS are connected through the main program script which directly interfaces with sensors. The collected data are stored in the laptop drive space which is accessed by the MATLAB program. For example, IMU data are stored in the memory space of Arduino at first then each data packet is transferred to a laptop with a serial communication tool that is connected via a USB port. The data is stored in a file and accessed by MATLAB.

2.9. Implementing and Training AI Models

Implementing and training the AI models involved AI model selection and training, and validation methods used.

2.9.1. Model Selection

Model selection of deep learning model for visual and audio analysis satisfies the requirements of analyzing the data features and patterns in visual and audio. Deep learning Layers (DLL) have classification and analysis layers that are suited for visual and audio data classification and prediction. Recurrent Neural Network (RNN) model is used for LSTM. LSTM is used to recognize sequences in data and predict their classification.

2.9.2. Training and Validation

The training and validation of CNN and LSTM involves data preparation, network design, and training that includes customizing the hyperparameters and regularization. This process is to prevent overfitting and exploding of gradients during training and improve performance and generalization.

2.10. Conducting Comprehensive Performance Testing

Conduction of comprehensive performance testing is performed with simulation and real-world testing and through feedback and improvement of NIFDT. This process involves defining performance metrics, data preparation, model evaluation, testing, tuning, and model comparison.

2.10.1. Simulation and Real-World Testing

It is critical to perform simulation and real-world testing on CNN and LSTM of NIFDT. The simulation testing will assess how well the model performs in a synthetic controlled environment with given input data. It is easily managed and controlled but it is simplified and has the problem of being overfit with simulated design and not general for real-world scenarios. So real-world testing would validate how well the system does in an unknown environment. It will show how well the simulated system works and adapts to the real world. But in this case, the real-world simulation comes with more complexity.

2.10.2. Feedback and Improvement

Feedback and improvement of NIFDT consists of creating performance metrics such as accuracy, false positives, scores, and loss analyzing the data, and keeping the training logs. Researching use cases and case studies of similar NIFDT will provide the needed user feedback to improve and solve the limitations.

2.11. Scope and Limitations

2.11.1. Non-Invasive Techniques and AI Tools

This proposed solution is embedded with a hybrid of Google-net and LSTM for visual fault detection, Yam-net for audio fault detection, and Long-Short-Term-Memory (LSTM) for 3-axis vibration and gyroscopic fault detection.

Table 3. AI Tools

AI technique	Description
A hybrid of Google-net and LSTM (Hybrid-net)	Pretrained neural network of Convolution Neural Network (CNN). This is used to learn the features of faults that are recorded in visual data.
Yam-net	Like Hybrid-net, Yam-net is a pre-trained deep-learning neural network that can recognize 521 classes of audio events. This is used to detect fault noise in audio data.
Long Short-Term Memory (LSTM)	A version of Recurrent Neural Network (RNN) that could recognize long sequences of events and data. For example, language. This is used to detect fault features in sequences of 3-axis vibration and gyroscopic data.

2.11.2. Technologies Covered

Technologies covered would include the machine learning framework for CNN and LSTM layers in MATLAB tool for developing, training, and evaluation. This would include a range of frameworks, libraries, and tools designed to support the development of data collection features extraction, and decision-making. Tools such as deep learning toolbox, data management methods, compilation and downloading of firmware.

2.11.3. Applications Covered

The applications covered are image classification, object detection, image segmentation, video analysis, time-series anomaly detection, and prediction. Applications covered leverage the AI technologies and address complex problems that could be generalized and applied to diverse areas including healthcare, finance, entertainment, security, and robotics.

2.12. Limitations

The limitations of NIFDT would be the quality of the data. If the data is lacking, insufficient, or imbalanced then the system trained with the data will lack accuracy and prove to be less useful. Insufficient will produce a system that is either unreliable or overfitting and not generalized enough for real-world usage.

Also due to the nature of CNN and LSTM being AI tools, it is designed to operate as a black-box which lacks transparency as to what happens when input is computed into output. Each time is not explicitly shown while training and validating which would require confirming the output values to provide trustworthy results.

The hardware sensors data collected and analyzed are limited to its hardware capabilities. The noise and interference from surroundings could affect the quality of the data and the model’s performance. The techniques used in the sensors are embedded in the hardware itself and cannot be modified or improved. Low quality would decrease the accuracy of the collected data and regrade the NIFDT system which negatively increases false positives.

2.13. Challenges in Sensor Accuracy and Data Quality

It is necessary to analyze the challenges in sensor accuracy and data quality to consider the issues related to hardware limitations as this will significantly impact the performance of NIFDT. The AI tools and models rely heavily on the accuracy of the data which they process.

2.13.1. Sensor Limitations

Sensor limitations include surrounding noise and interference, hardware limitations, calibration issues, wear, cross-sensitivity, and limitations to resolution. If the hardware is low quality then the data collected would be less reliable and sensitive to surrounding noise which could be from other signals emitting nearby.

2.13.2. Data Quality Issues

The data quality issues consist of issues containing noise and interference, incomplete data, inconsistent data, inadequate preprocessing, and excessively high datasets that could lead to missed data and computational demands. These would reduce the accuracy of the collected data and result in poor data quality which would result in inaccurate fault detection and decreased system performance. Handling the issues would increase system complexity in handling noises, missing and inconsistent data, and eventually unreliable output.

2.14. Constraints in Fault Detection Capabilities

The constraints in fault detection and capabilities rely on the quality of the available data. The historical data is critical to extract needed features and patterns. A fault detection system would require extensive historical data for training and validation.

But in most cases, this historical fault data is sparse or non-existent which makes training very difficult to detect rare and complex faults and provide system scalability.

2.14.1. Detection of Rare or Complex Faults

Generalization to detect new, rare, or complex faults is limited to available data. AI models are trained on specific fault types but if there is insufficient data then it is difficult to generalize and predict unforeseen faults. This would result in a system that fails to detect and classify new fault types resulting in undiagnosed issues and system failures.

2.14.2. System Scalability

System scalability involves the ability of the system to handle increased loads, accommodate various datasets, and extend its capabilities to other fault detection scenarios without negative effects such as performance degradation or additional resource drain. The challenges to this would be managing the increased data and storage requirements, computational resources, increase in model complexity and training, complexity in system integration and maintenance, and latency in real-time processing which all could lead to system performance degradation and negatively impact the real-time fault detection.

2.15. Types of AI (Artificial Intelligence) Techniques

Machine learning has played increasing roles in heavy Industrial Automation (IA) with the emerging concept of Industry Standards which produce more efficient and effective prediction algorithms. This trend is expected to propagate to other related IA fields that could involve other kinds of motors and machine components. This research scope is in DC motor simulation using artificial neural networks.

There are different types of AI for different requirements and needs. But all need to be sufficiently and properly trained and learned. The “learning” part of machine learning attempts to minimize error and maximize the likelihood of their prediction being true. In this project, we want to model the fault prediction characteristics and collect sensory data before and after the DC motor faults. This collected data is organized supervised and provided to the AI algorithm for training.

As there are different forms of AI, there are different uses and applicable environments and needs. Such as; Predicitvity, Performance, and Prognostics. Performance by combining analytics and machine learning, to develop specific algorithms for optimizing processes with accelerated resolution times, for improved operational efficiency.

Prognostics to apply Neural networks, deep-learning, and reinforcement learning technologies, to forecast operational scenarios for managing risk, maximizing profitability, and improving sustainability.

There are different areas AI is used and applied. Industrial Automation is one of the major sectors where humans are surrounded by complex robots all around and AI would support its need for operation and maintenance to reduce financial burden by preventing abrupt mechanical failure that could lead to fatal accidents.

Being employed from workforce planning to product design, for improving efficiency, product quality, and employee safety. In factories, machine learning and artificial neural networks are employed to support the maintenance of critical industrial equipment, which can accurately predict asset malfunction. Robots are an integral part of the production process. With the use of AI, robotics can take instructions from humans and work productively alongside them.

So, there are various types of AI and Its algorithms to process different types of needs and requirements. Some would require human interaction to filter input data for proper training of the AI machine and there are deep learning algorithms that do not require human supervisory interaction. As shown below, we have Machine Learning (ML), Neural Networks (ANN), Robotics, Expert Systems, Fuzzy logic, and Natural Language Processing. We will focus on ML, ANN, and Fuzzy logic AI algorithms for our paper.

2.15.1. Machine Learning (ML)

ML (Machine Learning) is a type of AI that would learn from each phase with and/or without human supervisory interactions. At each phase, ML would learn how to process itself to get closer to the desired outcome. Below is an organizational chart that characterizes the ML into how much human interaction is needed. Largely, they are categorized into Supervised Learning, Unsupervised Learning, and Reinforced Learning.

2.15.2. Artificial Neural Network (ANN)

ANN is an Artificial Neural Network consisting of neurons that can be activated using different kinds of activation functions. Each step between the neurons is equipped with bias and weight values that would result in output results. If the resulting value is not met then the algorithm reverses the steps and adjusts the bias and weights and results with a black box model as a result. This would not require the user to compute the algorithm and ANN would create the algorithm by itself looking at the input and desired output.

2.15.3. Deep Learning (DL)

The below figure shows the difference between Machine Learning (ML) and Deep Learning. In general, Deep Learning is part of Machine Learning. Machine Learning is an Artificial Intelligence method using computer science and statistics to have a machine learn patterns in numbers and data and to output its way to compute predicted results. As explained above, this can be done with supervised and unsupervised data.

2.15.4. Machine Learning vs Deep Learning

Meanwhile, Deep Learning is processed without human interaction. It is modeled with a neural network similar to the human brain and its output is computed automatically through training by using weights bias values and activation functions on each neuron. As shown below, Machine Learning requires an extra step for the human interface to extract necessary features of the data for the machine to look and learn. As for Deep Learning, this is done by structuring neural networks and assigning necessary values. Because of the increased complexity of Deep Learning, more data is required for training to be able to predict reliable results.

2.15.5. Simulation Results on NASA Turbo Data using DCNN

Below is the result from the sample simulation and experiment with MATLAB and toolbox to process the collected data using DCNN.

The default input raw data is computed into MATLAB as a matrix and outputted into time domain data as shown below.

2.16. Case Studies

2.16.1. Survey of PdM on Common Fault Prediction with AI

There are several publications and research in the field of ML and PdM within the Automotive sectors, Industrial Automation, and other sectors that deal with heavy numbers of machinery with motors.

Most of the papers lacked the PdM model with data collection of sensory data from DC Motors and classification algorithm to provide the cluster of data to ML for pattern recognition. The cluster of motor harmonic power data of voltage and current relation to motor speed, temperature, and vibrations. I believe that concentrating on the spectrum data of motor power harmonics data with aftereffects of temperature and vibration would provide patterns that could support ML for better recognition of motor faults. As stated previously, there is too much data to collect from various types of motors and to have one algorithm to filter and map it to one model. So we provide the effects of the fault as a pattern for ML to recognize and scientifically recognize possible motor faults.

2.16.2. Current research state of PreM

Below is my summarized ongoing current research in the field of AI in the PdM area and their current progress. It is categorized into P as partially developed, D as already developed and N as not has started to implement. P = Partially developed or in progress, D = Developed, N = Not Started.

A general Predictive Model using Machine Language (ML) techniques. Identified proposed approaches and techniques. In addition, it is possible to remark that PdM itself emerges as a new tool for dealing with maintenance events. Industry 4.0 advancements and relation to PdM. (D)

Developed a Simulink model that considers inputs from motor variables. ANN is used to train and develop a model to predict motor fault probability. The model includes data collection, data characterization, and computation to predict results. (D)

Develop simulation proof that shows improved fault predictability with an enhanced data model that relates to data acquisition and ANN training models using harmonic spectrum in power and vibrations. (P)

Develop a Proof of Concept (PoC) of Motor Fault Predictability on a DC motor with vibration sensors and power sensors using ANN modeling and comparison of different AI algorithms and training. Live real-time calibration of the collected data for interactive training of AI for better predictability results. Deployment of the trained model into Arduino for real-time assessment of Motor Fault Predictability. (N)

Real-time feed of sensory data of power harmonics as computed spectrum data and harmonic vibration amplitudes for AI training. The AI would be trained with data to recognize fault patterns and predict the fault mode and faulty Electric Motor parts. (N)

Discuss and compare the PdM AI algorithm and training methods, with Pros and Cons (N)

My dissertation makes the following contributions:

Explores AI training and learning used in Electric Motor fault prediction referencing harmonic vibration and power consumption in spectrum values.

Outlines a complete cycle simulation framework model that collects live feed data, replays, trains, and observes results to deployment for verification.

Demonstrates a three-input layered training model with forty-eight neurons in the hidden layer that can provide a set of weights and biases in the trained PdM model. This framework is then used to train the AI using data collected using an Electric Motor equipped with an accelerometer and power monitoring sensors.

2.17. Motor Fault Detection Case Study

The DC motor is comprised of several moving parts that are dedicated to sudden moves and changes such as vibration. Below are the parts of the DC motor. Each part is an essential component of the DC Motor that moves the motor. Parts include Brushes, Contacts, Commutator, Windings, Axle, Magnets, Bushing, and Can.

Each component inside the motor must operate in a perfectly harmonious way for it to work. The difficult part of it is when a small component inside starts to degrade or show wear. This could lead to difficult intermittent issues to diagnose and detect and because motors are crucial components in a large-scale factory line, if one small motor fails abruptly then the whole factory could come to a sudden halt or even worse could hurt the nearby workers. Therefore, there are models and methods implemented for maintenance, especially in industrial factories and automation plants. This paper will focus on automating and predicting possible future faults using Machine Language so such loss can be avoided.

We will study and investigate AI training algorithms and methodologies used for PdM. Data are sample data from real Electric Motors from private and public government sites such as NASA. The data are fed into MATLAB tool as a vector or table and AI trained and results are compared with random data to show the predictability.

2.18. Detecting Motor Fault Patterns from Literature

The data used for the simulation is given by the findability, accessibility, interoperability, and reusability (FAIR) Act webpage portal [4].

My proposal used this open dataset due to not having access to industrial-quality motors and hardware. My paper took advantage of the concept of FAIR and support allowed me to get reliable datasets so that we can try different algorithms. The field of predictive maintenance is particularly challenging in this aspect as most don't have access to full-size industrial equipment or there are no available datasets representing rich information content in different evolutions of faults.

2.19. Fault Detection Examples

2.19.1. Illustrative Examples and Error Analysis and Types of Errors

The collected data is compiled into the below graphs that show abnormal motor operation and describe the fault condition indicators. The indicators are shown with red circles.

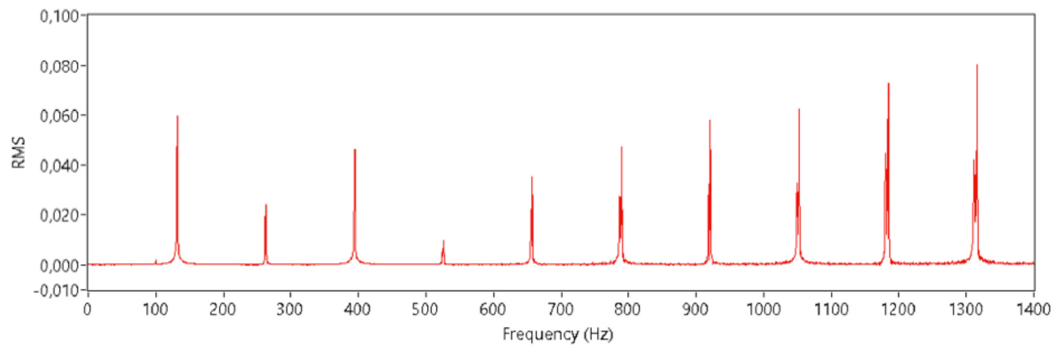


Figure 1. Sample Power Spectral Density [4]

Below are data collection and conditioning viewed in a plotted graph. This focus clustered of collected data is current, voltage, speed, temperatures in ambient and motor.

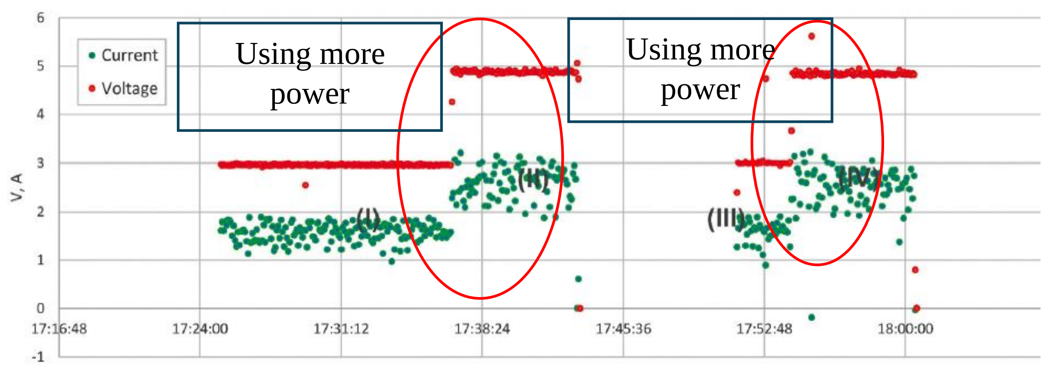


Figure 2. Sample Current and Voltage [4]

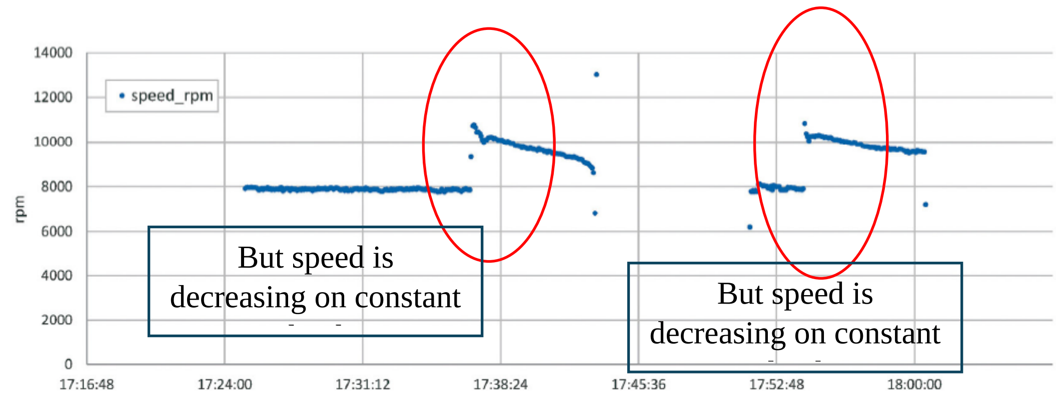


Figure 3. Sample Speed RPM [4]

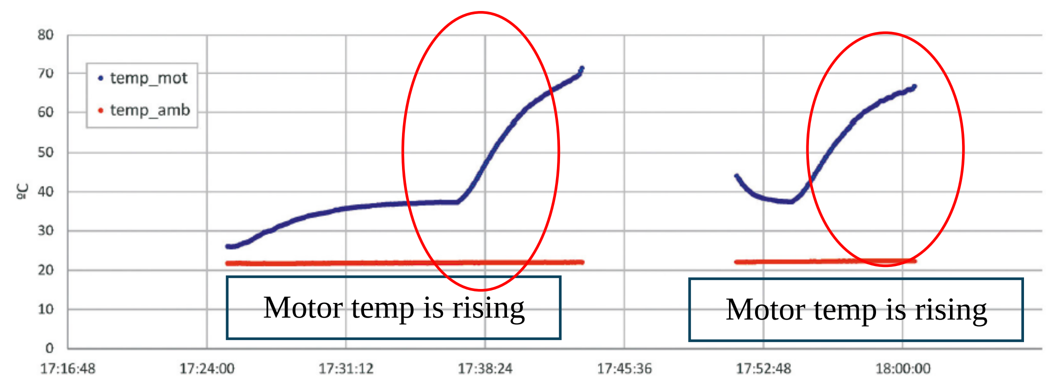


Figure 4. Sample Motor and. Temperature [4]

As you can see there is a sudden spike in DC Motor speed as there is a sudden decrease in DC Motor voltage, this could be a sign of possible intermittent fault. The DC motor rotation speed signal spiked possibly due to an electrical component fault as there was a sudden voltage drop or mechanical issue that pulled a sudden load.

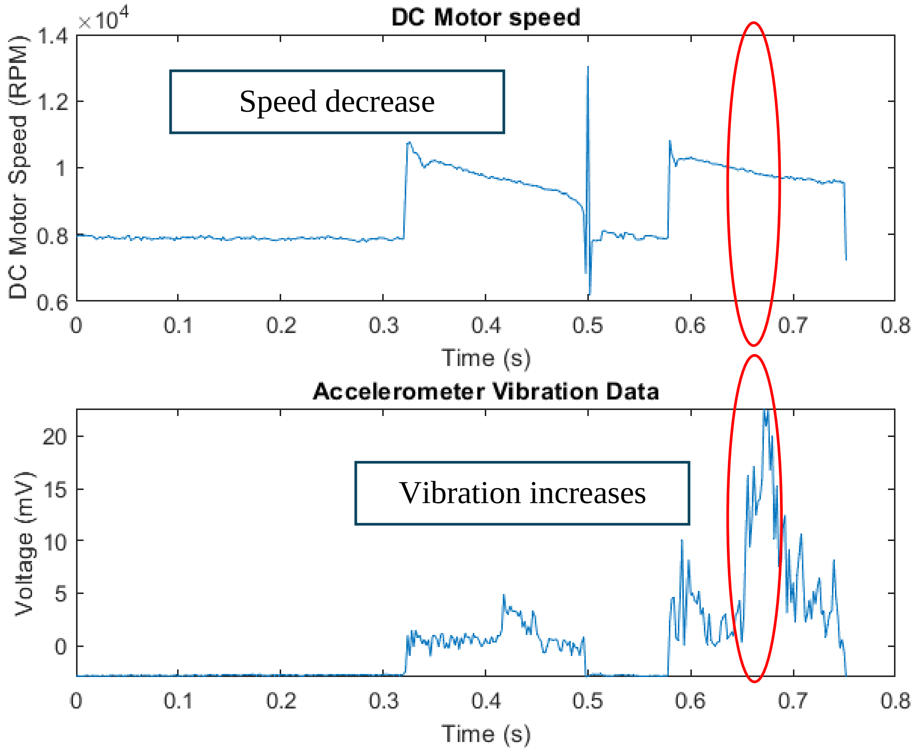


Figure 5. Sample Motor Speed and Vibration data [4]

The above-compiled graphs show wear to the motor as they were collected during the experiment. A large amount of data is collected and compiled to produce the above graphs Figures show obvious signs as you can see as the current and voltage increased but the speed decreased. This could mean there is an issue with a motor operation such as unwanted friction, or damaged components, due to vibration and/or heat which will decrease the motor efficiency and complete shutdown if not taken care of. I propose to use DL to find complex patterns and train to recognize them and produce accurate fault prediction in percentage.

CHAPTER THREE

TECHNIQUES AND TOOLS IN SIMULATIONS AND EXPERIMENTS

3.1. Research Design

The design of the research is on finding the technique and tools that would synthesize the data from various sensors and formulate well-rounded information so that this system users could make a proper decision. So the design consists of researching methodology, tools, and techniques used, implementing them, and collecting data. Afterward, we could verify the data and draw up conclusions on the research results.

The vast amount of sensor data from ambient and inside Electrical Machines are collected, analyzed, structured, and fed to AI for training and recognition. As mentioned previously, we will study many different types of AI training and algorithms and investigate the varieties in the current research area.

3.2. System Development

The system is developed with the fusion of various tools and software to improve the fault detection system. The development consists of researching the techniques and tools used in fault detection and applying the AI tools and software to manage the external sensors and their data for decision-making.

3.3. System Architecture Design

3.3.1. Overview

The overview of system architecture is designed to develop a combined system of AI tools and various sensors from outside the system. The idea of a non-invasive system is to have an independent fault detection system that manages the data from the outside point of view, like an observer. This way the combined output of the system can be monitored rather than the individual part. This idea is reflected in the design of this system architecture.

To monitor and detect features and symptoms of visual data, it is necessary to develop an AI tool that could sequence the extracted visual features. For this, a hybrid neural network that is CNN combined with RNN is developed and implemented to monitor subtle signs of visual faults.

3.3.2. Hybrid Neural Network Development

The Hybrid network involves the preparation of both CNN and RNN for combination. Both neural networks need to be of the same neural network architecture and have the same training model. Before connecting the two neural networks, it is necessary to have a fully trained CNN (Googlenet). This CNN would be trained with images of the vehicle’s instrumental panel with features and symptoms of different faults. After training, the network could recognize and properly classify the faults with a probability of over 85%.

The below figure depicts the first process of modifying the end set of layers to prepare concatenation with RNN. The layer labeled with 140 is modified as a continuation layer to cascade into the time sequencing layer of RNN.

133	'inception_5b-relu_5x5_reduce'	ReLU	ReLU
134	'inception_5b-5x5'	2-D Convolution	128 5×5×48 convolutions with stride [1 1] an
135	'inception_5b-relu_5x5'	ReLU	ReLU
136	'inception_5b-pool'	2-D Max Pooling	3×3 max pooling with stride [1 1] and paddin
137	'inception_5b-pool_proj'	2-D Convolution	128 1×1×832 convolutions with stride [1 1] a
138	'inception_5b-relu_pool_proj'	ReLU	ReLU
139	'inception_5b-output'	Depth concatenation	Depth concatenation of 4 inputs
140	'pool5-7x7_s1'	2-D Global Average Pooling	2-D global average pooling
141	'pool5-drop_7x7_s1'	Dropout	40% dropout
142	'loss3-classifier'	Fully Connected	1000 fully connected layer
143	'prob'	Softmax	softmax
144	'output'	Classification Output	crossentropyex with 'tench' and 999 other cla

Figure 6. CNN Poolset Modification

Below is the visualization of the process to combine the CNN with RNN. The order of the network represents how the data flows from the top block CNN to the bottom connected block of RNN.

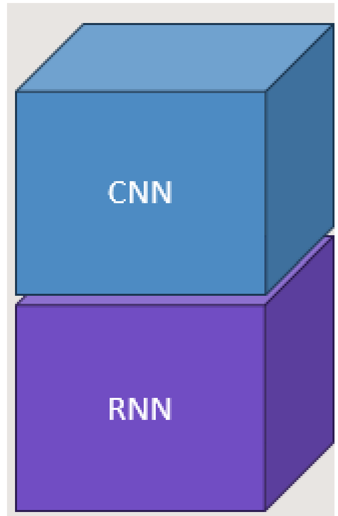
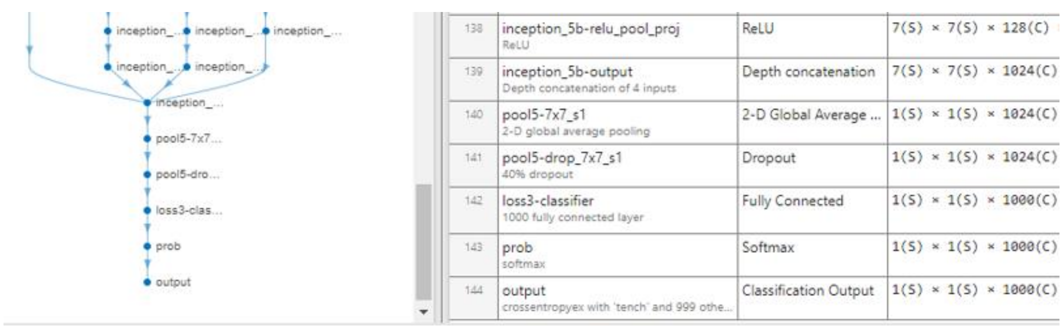


Figure 7. Hybrid Neural Network Block Diagram

The CNN network architecture and flow from input to output are shown on the left side of the figure and the details of the layers are shown on the right below figure. It consists of 144 layers with Classification Output of True or False which indicates the features in input are found with a match.



138	inception_5b-relu_pool_proj ReLU	ReLU	7(S) × 7(S) × 128(C)
139	inception_5b-output Depth concatenation of 4 inputs	Depth concatenation	7(S) × 7(S) × 1024(C)
140	pool5-7x7_s1 2-D global average pooling	2-D Global Average ...	1(S) × 1(S) × 1024(C)
141	pool5-drop_7x7_s1 40% dropout	Dropout	1(S) × 1(S) × 1024(C)
142	loss3-classifier 1000 fully connected layer	Fully Connected	1(S) × 1(S) × 1000(C)
143	prob softmax	Softmax	1(S) × 1(S) × 1000(C)
144	output crossentropyex with 'tench' and 999 othe...	Classification Output	1(S) × 1(S) × 1000(C)

Figure 8. CNN Layer Details

The RNN network is trained with features of vehicle faults and has time-sequenced information. It has 6 layers and a classification class of vehicle fault ASD

(Fault A) and other 4 faults. Its architecture and process flow is shown on the left and the layer detail information is shown on the right in the bottom figure.

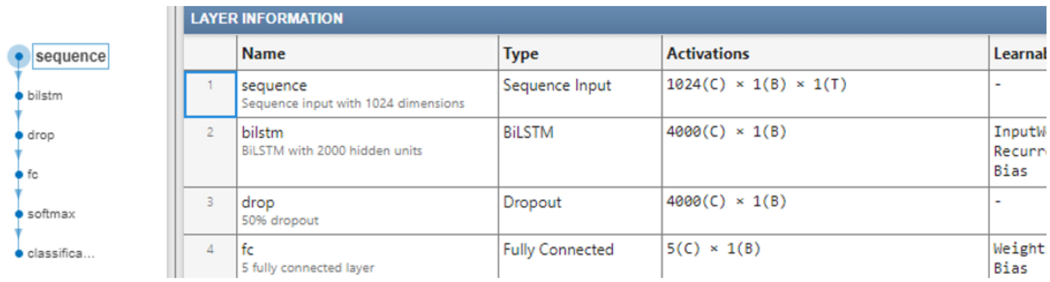


Figure 9. RNN Layer Details

By modifying the bottom layer of CNN and connecting it to the top layer of RNN, the resulting architecture of the hybrid is shown below. The pooling level 5 is connected from pooling level 4 with the inception process which is the intellectual method to extract features from complex patterns. The architecture of the hybrid neural network in the end portion is shown below.

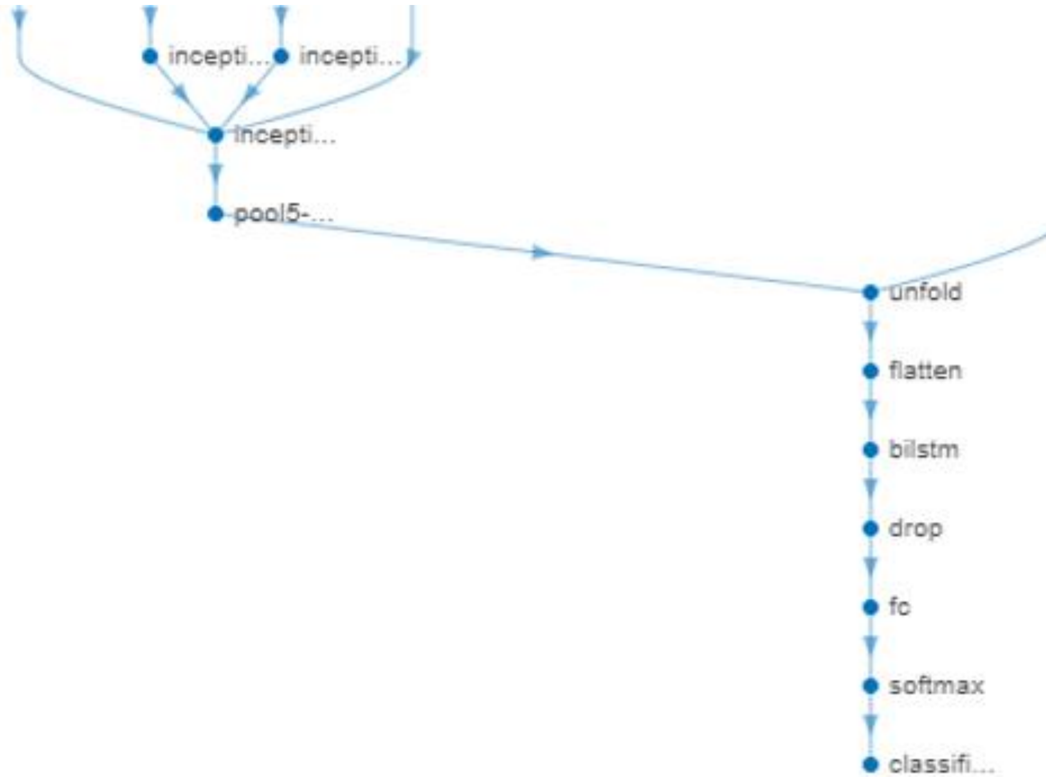


Figure 10. Hybrid Network Architecture

The details of the Hybrid Network Architecture are shown in the figure below.

148 classification
crossentropyex with 'BadASD' and 4 other classes

138	inception_5b-pool_proj 128 1×1×832 convolutions with stride [1 1] and padding	2-D Convolution	7(S) × 7(S) × 128(C) × 1(B)	Wei… 1 × 1 × 832… Bias 1 × 1 × 128	-
139	incep ReLU				
140	incep Depth				
141	pool5-7x7_s1 2-D global average pooling	2-D Global Average ...	1(S) × 1(S) × 1024(C) × 1(B)	-	-
142	unfold Sequence unfolding	Sequence Unfolding	1(S) × 1(S) × 1024(C) × 1(B) × 1(T)	-	-
143	flatten Flatten	Flatten	1024(C) × 1(B) × 1(T)	-	-
144	bilstm BiLSTM with 2000 hidden units	BiLSTM	4000(C) × 1(B)	InputWei… 16000 … Recurren… 16000 … Bias 16000 …	HiddenSt… 4000 ×… CellState 4000 ×…
145	drop 50% dropout	Dropout	4000(C) × 1(B)	-	-
146	fc 5 fully connected layer	Fully Connected	5(C) × 1(B)	Weights 5 × 4000 Bias 5 × 1	-
147	softmax softmax	Softmax	5(C) × 1(B)	-	-
148	classification crossentropyex with 'BadASD' and 4 other classes	Classification Output	5(C) × 1(B)	-	-

Figure 11. Hybrid Layer Details

3.3.3. Initial Experiment Model

The first stage of simulation involved collecting data from the motor and simulating the motor current and speed onto a graph. The simulated motor current which is power and output which is speed is shown on the graph. The graph shows an axis with motor current and an axis with motor speed. This is a ratio graph. The graph has two yellow lines to indicate the area within that is no fault. The other areas outside of this area bounded by the yellow lines are considered faults.

The simulation model consists of making the motor data and the graph and then taking a capture of the image and training the AI tool CNN. The training consists of images of the graph when there is a fault and when there is no fault. After training, we could expect the trained CNN to recognize the features in the graph to decide whether the image shown on the screen is faulty or not. Once the trained CNN is verified to be working properly and can recognize the difference between fault and non-fault graphs, we would move to collect actual sensor data and validate and test the trained CNN. Below is the flow of simulation and experimentation.

3.4. Experimental Setup with AI tools

The experimental setup involves applying different sensors with respectful AI tools. This is to match the functionality of the AI tool to the proper sensors. For example, cameras are installed to capture images and this would require an AI tool that specializes in image analysis. For in our case, we have adapted a Convolution Neural Network (CNN) for detecting and recognizing the features in the captured images.

Visual data from faults are recorded with a camera and used to train the Hybrid of GoogleNet with LSTM and output VisualFaultNet (VisFN). Trained VisFN can recognize the features and image characteristics shown on the recorded vehicle instrumental panel. All recorded faults data are recorded in the faults folder shown on the left side and the rest shows the process in the figure below.

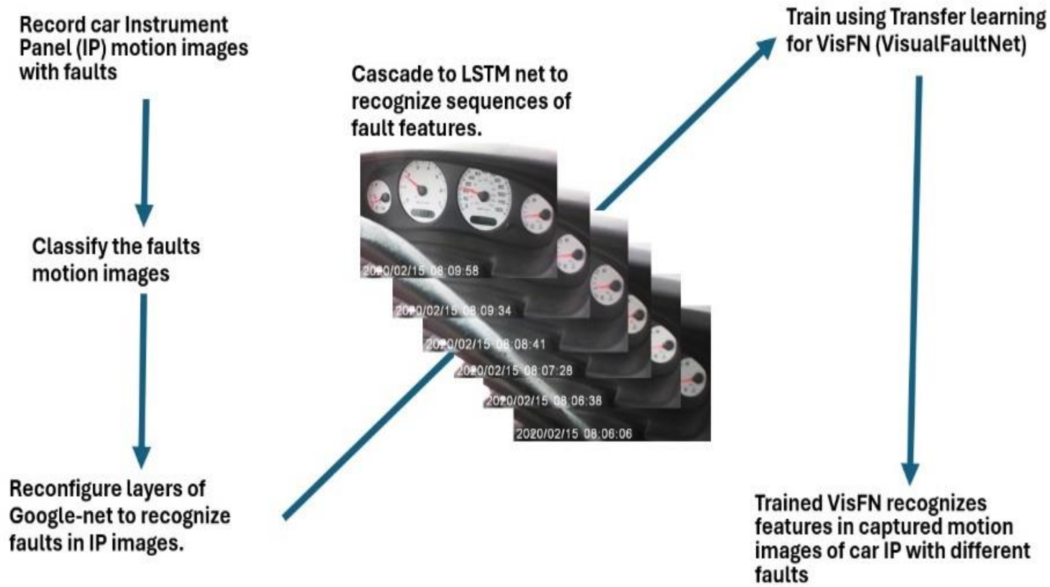


Figure 12. Visual Fault Detection of Instrument Panel

Audio data from faults are recorded with a microphone and used to train Yam-net and output AudioFaultNet (AudFN). A trained AudFN can recognize the features and sound characteristics recorded from vehicle cabins. All recorded faults data are recorded in the faults folder shown on the left side and the rest shows the process in the figure below.

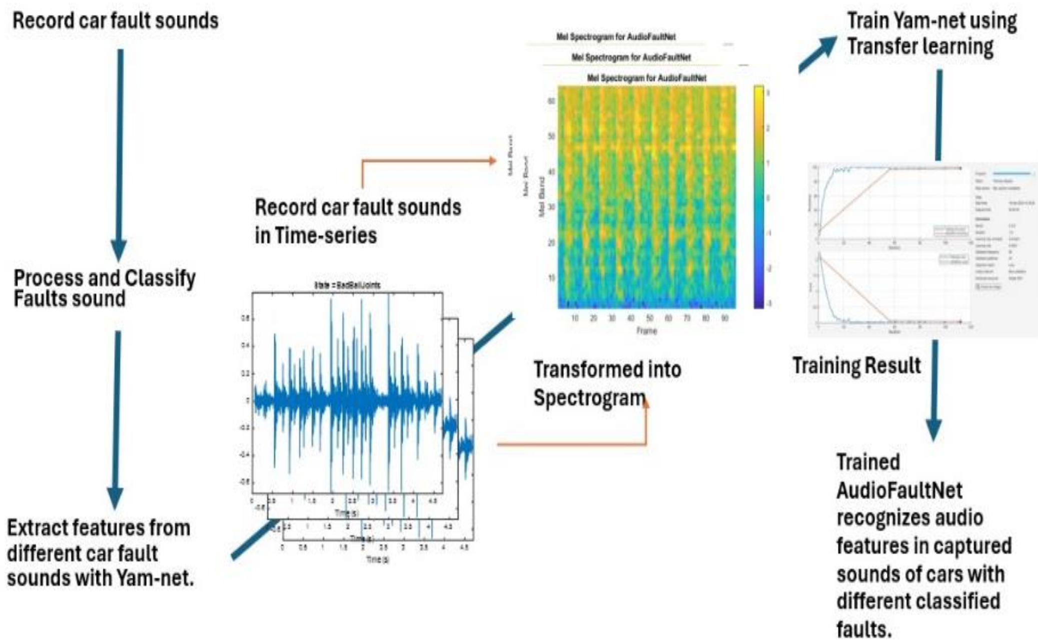


Figure 13. Audio Fault Detection with Microphone

3-axis data from faults are recorded with an IMU and used to train LSTM to output VibFaultNet (VibFN). Trained VibFN can recognize the features and 3-axis values recorded from vehicle chassis. All recorded faults data are recorded in the faults folder shown on the left side and the rest shows the process in the figure below.

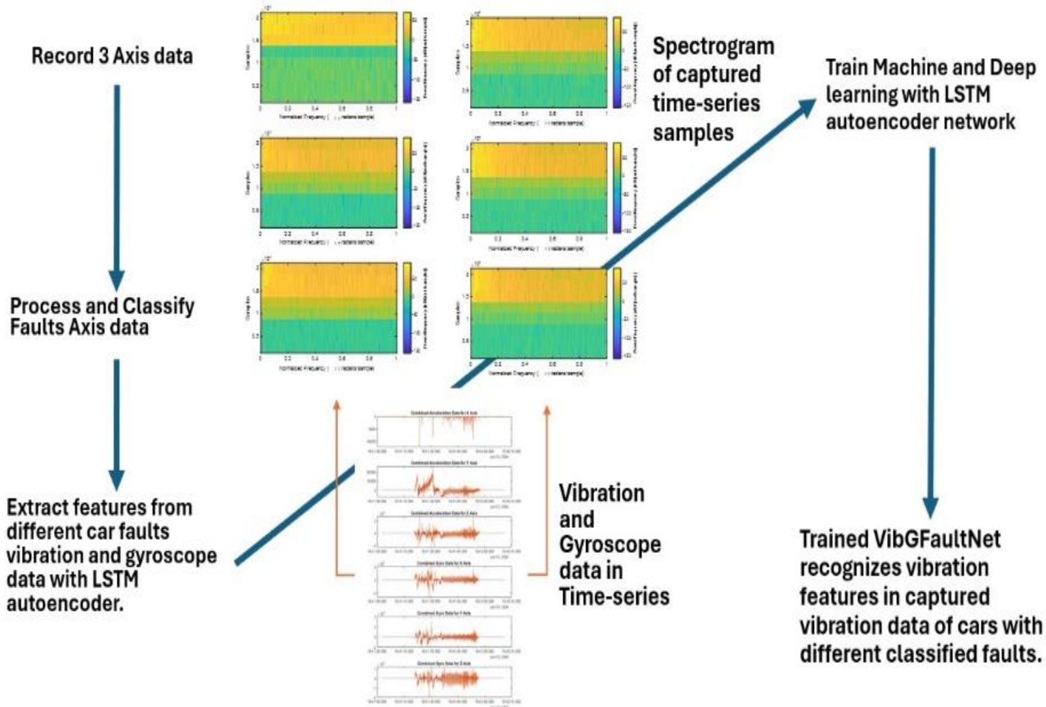


Figure 14. 3 Axis Fault Detection with IMU

The VisFN is a Hybrid-net classifier that uses a trained convolutional neural network (CNN) to process the video frames from the camera looking at the instrument panel and recognize and classify a fault that may show up in the gauges. If fault A shows up, it is labeled as Ac; if B shows up then Bc; and Cc, and so on.

The AudFN is a YamNet classifier that uses CNN to process the audio signals from the microphone listening to the sound from the car and recognize and classify a faulty noise that may show up. If fault A shows up, it is labeled as Am; if B shows up then Bm; and Cm, and so on.

The VibFN is an LSTM classifier that uses a Recurrent Neural Network (RNN) to process the vibration signals from the IMU monitoring to the vibration from the car and recognize and classify a vibration fault that may show up. If fault A shows up, it is labeled as Ai; if B shows up then Bi; and Ci, and so on.

This study considers the outputs Ac, Bc, and Cc to be confidence values taking on real numbers between zero and one. That is Ac, Bc, Cc,…∈ $[0,1] \subset \mathcal{R}$. Similarly, Am, Bm, Cm ∈,… $[0,1] \subset \mathcal{R}$ and Ai, Bi, Ci,…∈ $[0,1] \subset \mathcal{R}$. Figure 7 shows where these as the outputs of the fault classifiers.

Below is the flow of the decision-making from gathering sensor data from the left to assessing the collected prediction scores and using fuzzy logic to output the most likely fault that is shown on the right of the figure below.

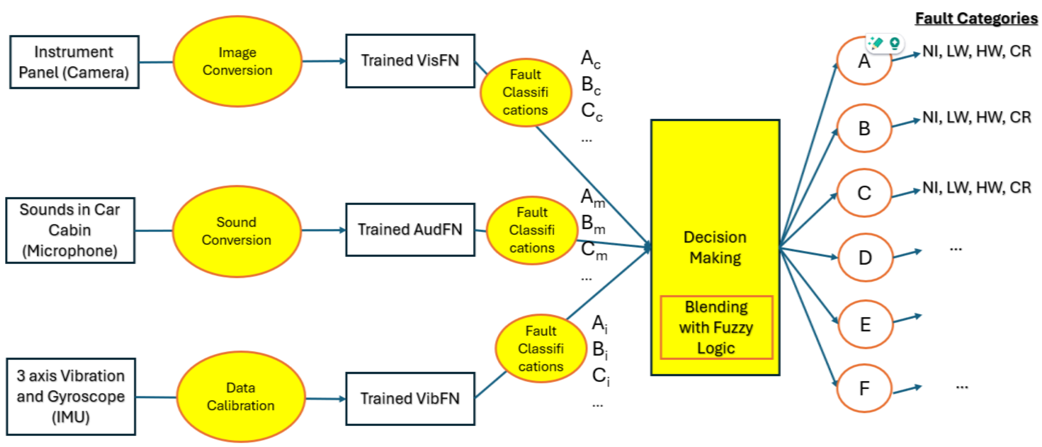


Figure 15. Decision-making with FIS

The table below shows a more elaborate list of faults and classification outputs. Includes all sensors investigated that could be used to detect faults in the environment and provide useful data to determine the true fault. The list consists of a Camera, Microphone, IMU, Thermistor, Barometer, Gas sensor, Humidity sensor, Proximity sensor, and Yaw-rate sensor.

Devices	"A" Faulty Battery	"B" Faulty Alternator	"C" Faulty Starter	"D" Faulty Spark Plug	"E" Faulty Wheel	"F" Faulty Engine (Water Pump, EGR)
Camera	Ac	Bc	Cc	Dc	Ec	Fc
Microphone	Am	Bm	Cm	Dm	Em	Fm
IMU	Ai	Bi	Ci	Di	Ei	Fi
Thermistor	At	Bt	Ct	Dt	Et	Ft
Barometer	Ab	Bb	Cb	Db	Eb	Fb
Gas sensor	Ag	Bg	Cg	Dg	Eg	Fg
Humidity sensor	Ah	Bh	Ch	Dh	Eh	Fh
Proximity sensor	Ap	Bp	Cp	Dp	Ep	Fp
Yaw-Rate sensor	Ay	By	Cy	Dy	Ey	Fy

Figure 16. Classifications of Faults and Devices

Below is an example of the flow from inputs with the probability of real numbers between 0 (No) to 1(Yes). Each sensor will submit its probability score and the higher the score indicates its confidence. Example situations are described in Rules 1, 2, and 3 below. The result would indicate the computed fault prediction from the uncertainty values from the sensors.

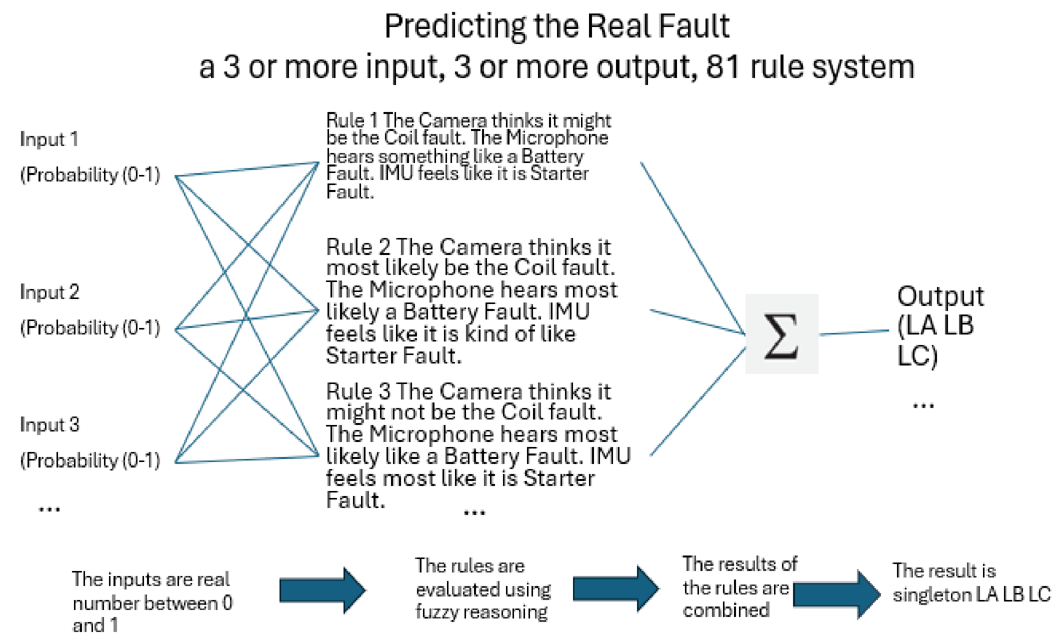


Figure 17. Fault Prediction Flow

3.5. Decision Making Logic.

Types of Faults:

A, B, C, D, E, G, …

Detection Devices:

c, m, i, t, b, g, h, p, y, …

Fault Variables Reported by Individual Devices:

Definition.

A_c □ variable for fault A reported by device c

A_m □ variable for fault A reported by device m

A_i □ variable for fault A reported by device i

$\vdots$

B_c □ variable for fault B reported by device c

B_m □ variable for fault B reported by device m

$\vdots$

C_c □ variable for fault C reported by device c

Value domain.

$$A_c,\ A_m,\ A_i,\ \cdots,\ B_c,\ B_m,\ B_i,\ \cdots,\ C_c,\ C_m,\ C_i,\ \cdots\ \in [0,1] \subset \mathbb{R}$$

Likelihood of Type of Fault based on Combined Reports from All Devices:

Definition.

$L_A \triangleq$ likelihood of fault A based on combined report from device $c, m, i, \cdots$
$L_B \triangleq$ likelihood of fault B based on combined report from device $c, m, i, \cdots$
$L_C \triangleq \cdots$
$\vdots$

Value domain

$$L_A,\ L_B,\ L_C,\ \cdots\ \in [0,1] \subset \mathbb{R}$$

Fuzzy Values and their Degree of Association or Membership Function for a Device:

Illustration

Suppose that x Is the value of the fuzzy variables of a particular device, and the fuzzy values for the variables are $\mu_{dev,low}(x),\ \mu_{dev,med}(x)\ \&\ \mu_{dev,high}(x)$.

As an example, further, suppose that.

$$\begin{aligned}
\mu_{dev,low}(x) &= \frac{1}{1+e^{-m_1(x-c_1)}} && m_1 = -15 && c_1 = 0.3 \\
\mu_{dev,med}(x) &= e^{-m_2(x-c_2)^2} && m_2 = 15 && c_2 = 0.5 \\
\mu_{dev,high}(x) &= \frac{1}{1+e^{-m_3(x-c_3)}} && m_3 = 15 && c_3 = 0.7
\end{aligned} \tag{1}$$

The fuzzy membership/degree of association function for the device is shown in the figure below.

If $x = 0.6$, then

$$\mu_{dev,low}(x) = 0.010087,\ \mu_{dev,med}(x) = 0.86071\ \&\ \mu_{dev,high}(x) = 0.18243 \tag{2}$$

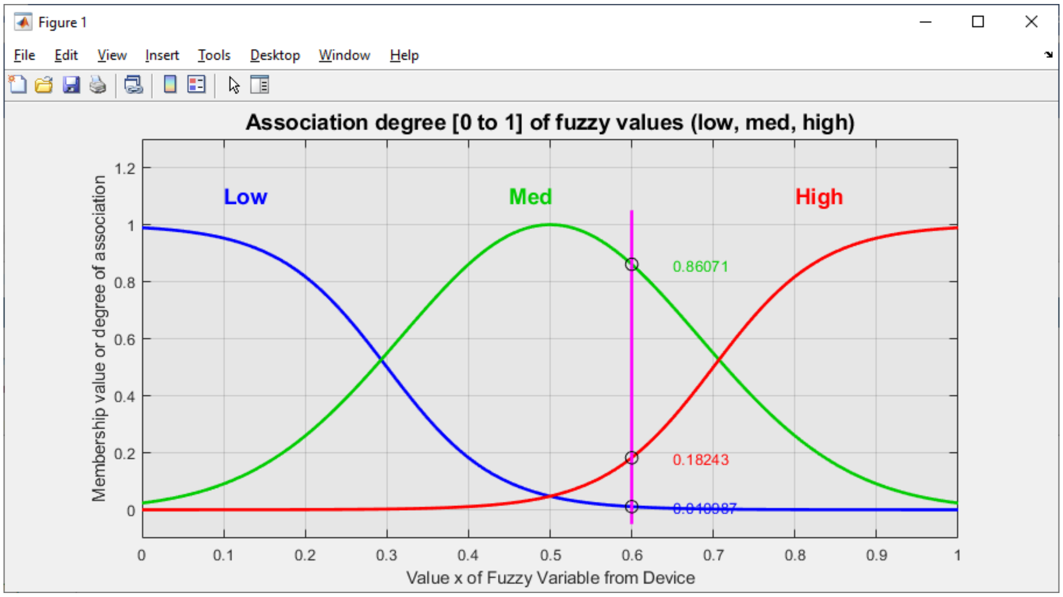


Figure 18. Fuzzy Values

Detailed list of fuzzy values of devices.

A detailed list of device values and their fuzzy membership functions is shown below.

$$
\begin{array}{l}
\mu_{A_c,low},\ \mu_{A_c,med},\ \mu_{A_c,low},\ \cdots \\
\mu_{A_m,low},\ \mu_{A_m,med},\ \mu_{A_m,low},\ \cdots \\
\mu_{A_i,low},\ \mu_{A_i,med},\ \mu_{A_i,low},\ \cdots \\
\vdots \\
\mu_{B_c,low},\ \mu_{B_c,med},\ \mu_{B_c,low},\ \cdots \\
\mu_{B_m,low},\ \mu_{B_m,med},\ \mu_{B_m,low},\ \cdots \\
\vdots \\
\vdots \\
\mu_{C_c,low},\ \mu_{C_c,med},\ \mu_{C_c,low},\ \cdots \\
\vdots
\end{array}
\qquad (3)
$$

Fuzzy Logic Table for Device Fault Level

Table 4. Table of Rules

	Camera report A_c	Microphone report A_m	IMU report A_i	Combined Likelihood L_A
Rule 1	$\mu_{A_c,low}$	$\mu_{A_m,low}$	$\mu_{A_i,low}$	w_1
Rule 2	$\mu_{A_c,low}$	$\mu_{A_m,low}$	$\mu_{A_i,med}$	w_2
Rule 3	$\mu_{A_c,low}$	$\mu_{A_m,low}$	$\mu_{A_i,high}$	w_3
Rule 4	$\mu_{A_c,low}$	$\mu_{A_m,med}$	$\mu_{A_i,low}$	w_4
Rule 5	$\mu_{A_c,low}$	$\mu_{A_m,med}$	$\mu_{A_i,med}$	w_5
Rule 6	$\mu_{A_c,low}$	$\mu_{A_m,med}$	$\mu_{A_i,high}$	w_6
Rule 7	$\mu_{A_c,low}$	$\mu_{A_m,high}$	$\mu_{A_i,low}$	w_7
Rule 8	$\mu_{A_c,low}$	$\mu_{A_m,high}$	$\mu_{A_i,med}$	w_8
Rule 9	$\mu_{A_c,low}$	$\mu_{A_m,high}$	$\mu_{A_i,high}$	w_9
…	…	…	…	…
Rule 19	$\mu_{A_c,high}$	$\mu_{A_m,low}$	$\mu_{A_i,low}$	w_{19}
Rule 20	$\mu_{A_c,high}$	$\mu_{A_m,low}$	$\mu_{A_i,med}$	w_{20}
Rule 21	$\mu_{A_c,high}$	$\mu_{A_m,low}$	$\mu_{A_i,high}$	w_{21}
Rule 22	$\mu_{A_c,high}$	$\mu_{A_m,med}$	$\mu_{A_i,low}$	w_{22}
Rule 23	$\mu_{A_c,high}$	$\mu_{A_m,med}$	$\mu_{A_i,med}$	w_{23}
Rule 24	$\mu_{A_c,high}$	$\mu_{A_m,med}$	$\mu_{A_i,high}$	w_{24}
Rule 25	$\mu_{A_c,high}$	$\mu_{A_m,high}$	$\mu_{A_i,low}$	w_{25}
Rule 26	$\mu_{A_c,high}$	$\mu_{A_m,high}$	$\mu_{A_i,med}$	w_{26}
Rule 27	$\mu_{A_c,high}$	$\mu_{A_m,high}$	$\mu_{A_i,high}$	w_{27}

Fuzzy Rules for Level Device Fault

Rule 1 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_1$

Rule 2 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_2$

Rule 3 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,low}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_3$

Rule 4 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_4$

Rule 5 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_5$

Rule 6 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,med}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_6$

Rule 7 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_7$

Rule 8 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_8$

Rule9 If A_c is $\mu_{A_c,low}$ and A_m is $\mu_{A_m,high}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_9$

Rule 10 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{10}$

Rule 11 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{11}$

Rule 12 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{12}$

Rule 13 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{13}$

Rule 14 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{14}$

Rule 15 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{15}$

Rule 16 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{16}$

Rule 17 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{17}$

Rule 18 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{18}$

Rule 19 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{18}$

Rule 20 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{18}$

Rule 21 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,low}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{18}$

Rule 22 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{18}$

Rule 23 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{18}$

Rule 24 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,med}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{18}$

Rule 25 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,low}$, then $L_A = w_{18}$

Rule 26 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is $\mu_{A_i,med}$, then $L_A = w_{18}$

Rule 27 If A_c is $\mu_{A_c,med}$ and A_m is $\mu_{A_m,high}$ and A_c is, $\mu_{A_i,high}$ then $L_A = w_{18}$

Same Rule 1 to Rule 27 to Fault B and Fault C.

For Fault B example,

Rule 1 If B_c is $\mu_{B_c,low}$ and B_m is $\mu_{B_m,low}$ and B_c is $\mu_{B_i,low}$, then $L_B = w_{31}$

Rule 2 If B_c is $\mu_{B_c,low}$ and B_m is $\mu_{B_m,low}$ and B_c is $\mu_{A_i,med}$, then $L_B = w_{32}$

And so forth…

For Fault C example,

Rule 1 If C_c is $\mu_{C_c,low}$ and C_m is $\mu_{C_m,low}$ and C_c is $\mu_{C_i,low}$, then $L_C = w_{61}$

Rule 2 If C_c is $\mu_{C_c,low}$ and C_m is $\mu_{C_m,low}$ and C_c is $\mu_{C_i,low}$, then $L_C = w_{62}$

And so forth…

Fuzzy Math Computation for Fault Level A from Combining All Devices

Strength of each rule using multiplication arithmetic for the AND operator

$$
\begin{aligned}
s_1 &= \mu_{A_c,low} * \mu_{A_m,low} * \mu_{A_i,low} \\
s_2 &= \mu_{A_c,low} * \mu_{A_m,low} * \mu_{A_i,med} \\
s_3 &= \mu_{A_c,low} * \mu_{A_m,low} * \mu_{A_i,high} \\
s_4 &= \mu_{A_c,low} * \mu_{A_m,med} * \mu_{A_i,low} \\
s_5 &= \mu_{A_c,low} * \mu_{A_m,med} * \mu_{A_i,med} \\
s_6 &= \mu_{A_c,low} * \mu_{A_m,med} * \mu_{A_i,high} \\
&\vdots \\
s_{27} &=
\end{aligned}
\tag{4}
$$

Fuzzy Math Computation for Fault Level B from Combining All Devices strength of each rule using multiplication arithmetic for the AND operator

$$
\begin{aligned}
s_1 &= \mu_{B_c,low} * \mu_{B_m,low} * \mu_{B_i,low} \\
s_2 &= \mu_{B_c,low} * \mu_{B_m,low} * \mu_{B_i,med} \\
s_3 &= \mu_{B_c,low} * \mu_{B_m,low} * \mu_{B_i,high} \\
s_4 &= \mu_{B_c,low} * \mu_{B_m,med} * \mu_{B_i,low} \\
s_5 &= \mu_{B_c,low} * \mu_{B_m,med} * \mu_{B_i,med} \\
s_6 &= \mu_{B_c,low} * \mu_{B_m,med} * \mu_{B_i,high} \\
&\vdots \\
s_{27} &=
\end{aligned}
\tag{5}
$$

Fuzzy Math Computation for Fault Level C from Combining All Devices strength of each rule using multiplication arithmetic for the AND operator

$$
\begin{aligned}
s_1 &= \mu_{C_c,low} * \mu_{C_m,low} * \mu_{C_i,low} \\
s_2 &= \mu_{C_c,low} * \mu_{C_m,low} * \mu_{C_i,med} \\
s_3 &= \mu_{C_c,low} * \mu_{C_m,low} * \mu_{C_i,high} \\
s_4 &= \mu_{C_c,low} * \mu_{C_m,med} * \mu_{C_i,low} \\
s_5 &= \mu_{C_c,low} * \mu_{C_m,med} * \mu_{C_i,med} \\
s_6 &= \mu_{C_c,low} * \mu_{C_m,med} * \mu_{C_i,high} \\
&\vdots \\
s_{27} &=
\end{aligned}
\tag{6}
$$

Sugeno-style Inference output using the weighted average

$$
L_A = \frac{\sum_{k=1}^{27} (s_k * w_k)}{\sum_{k=1}^{27} s_k}
\tag{7}
$$

Similar computations will yield the Combined Fault Levels A, B, C, etc.

$$L_A,\ L_B,\ L_C,\ \cdots \tag{8}$$

3.6. IMU data verification using spring-connected board.

The acceleration data are verified and checked with the setup below.

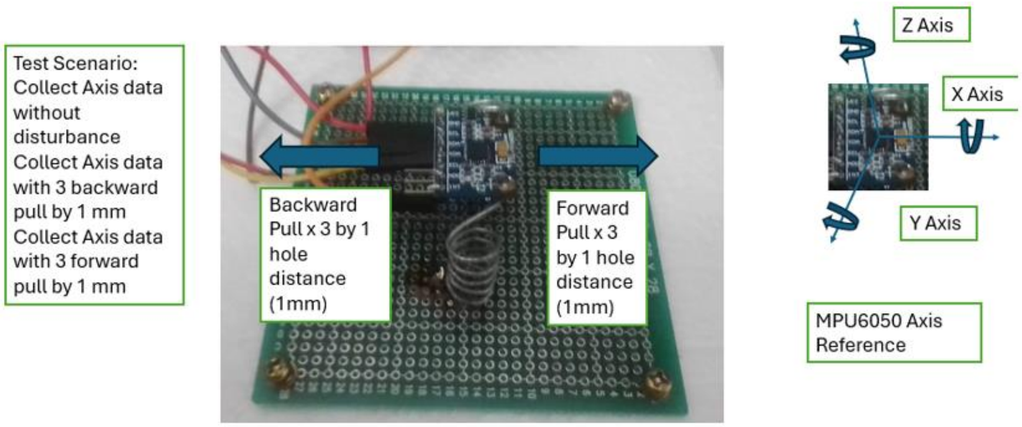


Figure 19. Setup for IMU Testing

The setup allows controlled forced disturbance to the IMU by pulling or pushing the IMU attached to the spring and board. The spring is soldered to the IMU and the board. This allows controlled testing by moving the IMU by measuring displaced distance by looking at the holes on the board. The IMU collects the data on the x-axis, y-axis, and z-axis. The sensor data are transmitted to Arduino which is connected and processed by MATLAB tool.

3.7. Frame image sequence validation test with hand gestures

The trained hybrid net is validated using captured video of hand gestures of one, two, and five. Different sequences of the three hand gestures are recorded with a web camera and provided to a hybrid net for classification. This test is done to verify and test how well the hybrid net could classify the correct sequence of the hand gestures. This test is performed first with a recorded video file and then tested with a live sequence. The live sequence included hand gestures from different angles as well which the hybrid net had a difficult time classifying correctly. In overall, the classification had approximately 70% success rate of correctly identifying the hand sequence and less than 60% from live video feed.

Below is a screenshot photo of two, five, and one from left to right.

Figure 20. Hand Gesture 251

3.8. Experiments with vehicle

3.8.1. Experiments Hardware Components

This system would have visual, and audio sensors and devices that will record a combination of ambient data. A combination of 9 different sensors is embedded to monitor the ambient machine surroundings as shown below.

Table 5. Devices

Device	Description
Camera	Device to record visual data
Microphone	Device to record audio data
Inertial Measurement Unit (IMU)	Device to record acceleration and gyroscopic data
Thermistor	Device to record temperature data
Barometer	Device to record pressure data
Gas sensor	Device to record gas chemical data
Humidity sensor	Device to record humidity data
Proximity sensor	Device to record surrounding proximity data
Yew-Rate sensor	Device to record gyroscopic rate data

To acquire data, we used different equipment and hardware. For example, we used the camera to acquire visual data from the Instrument Panel. The images are collected as frames to acquire the motion of the images rather than a still image. This would allow the CNN to acquire the features of the image but by combining it with LSTM, we are analyzing the sequence of the features and thus can recognize the animated frames. This is crucial to detect visual faults that are shown on the IP.

The Processing Unit used in the simulation is an Intel Processor i3 with 8 gigabytes of RAM. This processing unit is the factor in how well the simulation is run and how the AI program can run without any issues.

For devices that are difficult to attach to a personal computer, we used Arduino Mini to support gathering data through SPI connections. For example, the IMU06050 which is embedded with accelerometer and gyroscope hardware does not have the hardware capable connection to a personal computer. So we used an Arduino device to connect with this IMU06050 through SPI. The Clock and SPI connections are made with 0.5 breaker board cables. The processing unit is provided with 5V from the Arduino that is fed through from the personal computer.

The user interface used is the application that interfaces with the Arduino device. This provides the development interface and debugging functionality so that we can program the necessary codes and functions to acquire the axis data. This data is then formatted and offset to the Arduino device and then to the personal computer.

Other user interfaces are supported through MATLAB. MATLAB provides the necessary interface to script and code the necessary programs to train, simulate, and run the data collection for visual and audio data.

3.9. Vehicle setup with hardware

Devices are installed on the vehicle to record visual, audio, and 3-axis data. The camcorder is installed on the sun visor of the driver's side of the vehicle. This device will record the images of the instrument panel and sounds. The location is shown below.

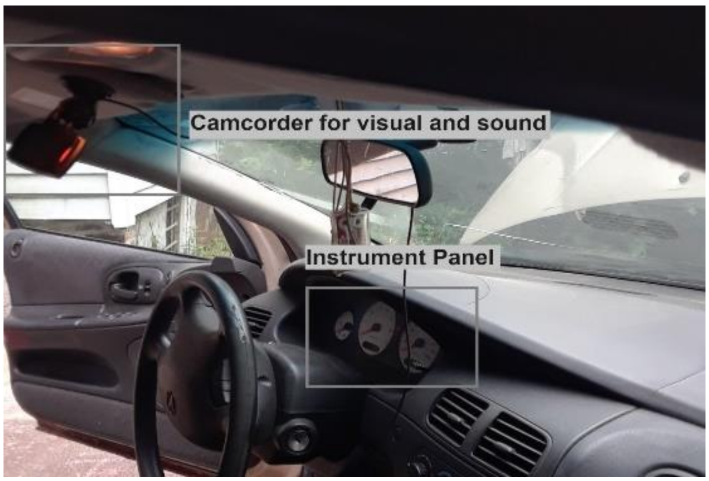


Figure 21. Image of Car Camcorder recording Instrument Panel and Sounds

The devices are installed to collect the necessary data that will be used for AI tools training. The real data collected is tested against the trained AI tools and check how proficient and what the success rate would be.

Below is the setup in the vehicle engine bay.

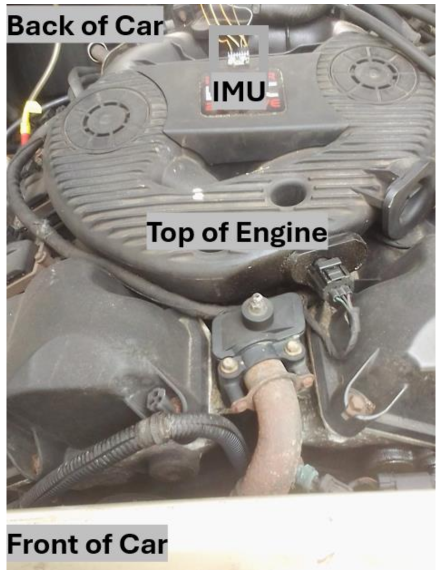


Figure 22. IMU in Engine Bay

The IMU is attached to the plastic holder on top of the engine. The vibration that is resonating from the engine is transferred to IMU and the recorded displacement in the axis, y-axis, and z-axis are recorded and shared with the Arduino board is that connected. Eventually, the real-time feed is passed to the connected laptop and used to train the AI tools and test for fault classification.

3.10. Vehicle starting issue scenario

There are 5 vehicle starting fault experimental cases.

3.10.1. Experiments Cases

The experiment cases of A, B, and C produce almost identical visual, sound, and vibrations that are difficult to detect. This experiment is to show the results of NIFDT detecting the features in image sequences, sound spectrum, and vibration data sequences and predict the fault using FIS. Below are descriptions of starting fault scenarios.

The car does not start due to “A” faulty Automatic ShutDown (ASD). The vehicle is installed with a damaged ASD relay. The relay is responsible for circulating the needed power to components inside the vehicle. Not having the relay installed properly in the vehicle prevents the vehicle from properly starting.

The car not starting due to “B” faulty battery. The vehicle is installed with a damaged and depleted battery. It is charged using cables to the car battery to 13V before experiments.

The vehicle failed to start with a “C” faulty coil. The vehicle is not installed with a coil fuse which will prevent the vehicle from starting. The coil is the vehicle component that converts the 12V battery to high voltage. This high voltage allows the necessary voltage for the spark plugs. The sparks lead to a high explosion inside the engine that is necessary to push the pistons and start the vehicle. So by not having the high voltage, the vehicle will not start.

The car does not start due to “D” faulty fuel injection. The vehicle relay for the fuel pump is removed. This will prevent fuel from being supplied to the engine. The car will start on the first attempt and will immediately halt.

The car not starting due to “E” faulty starter. The vehicle relay for the starter is removed. This will prevent the starter from engaging with gear.

Below is the table with 5 different starting faults that are created by removing the good battery, relays, and fuses in the vehicle and having the vehicle cranking and trying to start the vehicle.

Table 6. Vehicle Starting Fault Classifications

Devices	“A” Faulty ASD	“B” Faulty Battery	“C” Faulty Coil	“D” Faulty Fuel	“E” Faulty Starter
Camera	A_c	B_c	C_c	D_c	E_c
Microphone	A_m	B_m	C_m	D_m	E_m
IMU	A_i	B_i	C_i	D_i	E_i

Below is a photo of the bad battery that is depleted and needs to recharge to at least 13V to collect failing to start due to a bad battery fault. The healthy battery and connected vehicle parts need to be removed and disconnected before charging the bad battery. This needs to be done each time to collect the data for battery faults.

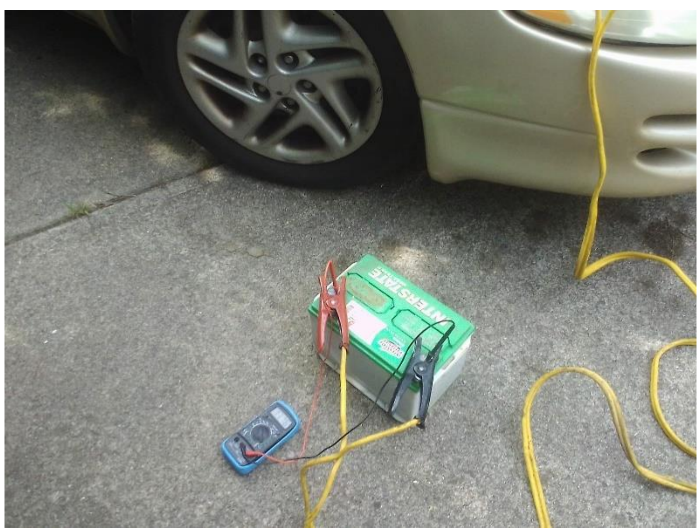


Figure 23. Vehicle Battery Setup

3.11. Vehicle wheel issue scenario using a surrogate model

3.11.1. Surrogate model

A surrogate model provides benefits to evaluating complex systems that could be computationally intensive, costly, or dangerous. This proposed model is a simplified part of the vehicle that is ratioed to reflect the dynamic properties of a wheel. This paper will explore, simulate, and experiment by introducing disturbance and issues to AC motors connected to peripherals, as a surrogate model for a vehicle’s wheel to capture the dynamic behavior approximating the system's behavior when different faults occur. This surrogate model can then be used to efficiently simulate and analyze the impact of various faults on vehicle wheels.

This model is proposed to avoid complications and safety concerns with creating possibly dangerous faults to the wheel of a moving vehicle on public roads. The features of the fault are monitored and data is collected using three sensors. The Instrument Panel (IP) of the vehicle is simulated with the engine’s Revolutions Per Minute (RPM) and speed (mph). This visual is provided to visual control hybrid AI-trained Convolution Neural Network (CNN) and Long Short Term Memory (LSTM). The sequence of visual features and patterns associated with the four different wheel-related faults are analyzed and learned. The sound and vibration data are collected from the motor using a microphone and IMU and provided to CNN for audio and LSTM for vibration, to process the features in the data and recognize the fault data patterns.

As the motor will display the simplified version of the vehicle between the engine to the wheel, this will provide the necessary vehicle wheel dynamics to understand the key parameters affecting the wheel’s movements affected by the four faults. After the dynamic features of the wheel affected by the faults are collected and trained, the combined AI tools will be able to recognize the faults with visual, audio, and vibration data and predict the fault. This paper will examine the methods of scalability of using AC motors to surrogate the dynamic motion of a vehicle’s wheel.

3.11.2. Fault Detection with AI tools

AI-driven fault detection methods and experiments are researched and studied for benefits and practicality. It offers merits compared to traditional fault detection methods of interval maintenance and replacement when broken. This paper proposes non-invasive detection methods with sensors installed external to the operational system. This way, the operational system is not modified and interrupted.

However, because the sensors are installed externally to the operational system and will monitor large and complex data from sensors and not from a single part of the system, this becomes difficult [3]. For example, the audio and vibration data collected are not from a single part of the system from inside but combined data with several parts. This makes the subtle telltale faults significantly more complex to analyze and predict. To predict the faults, this paper proposes three AI techniques.

3.12. Surrogate Model and Faults

3.12.1. Surrogate Model

The surrogate model will simulate a vehicle's wheel motion and rotation properties. The motor will act as the vehicle engine to the wheel and the round container attached is the vehicle wheel. For illustrative purposes, the motor surrogate model for the vehicle wheel is shown below.

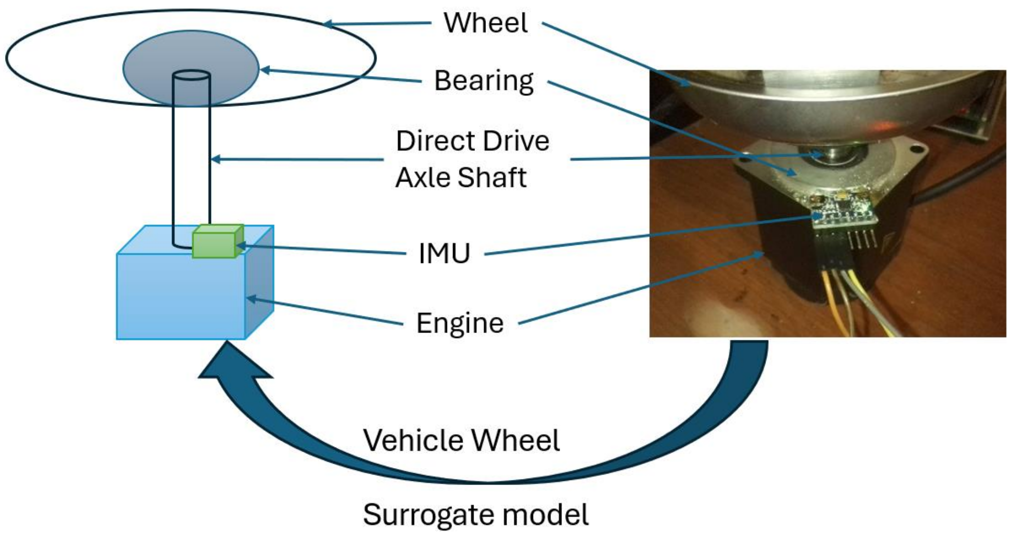


Figure 24. Motor Surrogate Model

This consists of an AC Motor connected to a round container, a Motor controller board, and a Motor inverter board. The AC motor is powered and controlled with a controller board and inverter board.

The setup of the AC motor and boards is shown below.

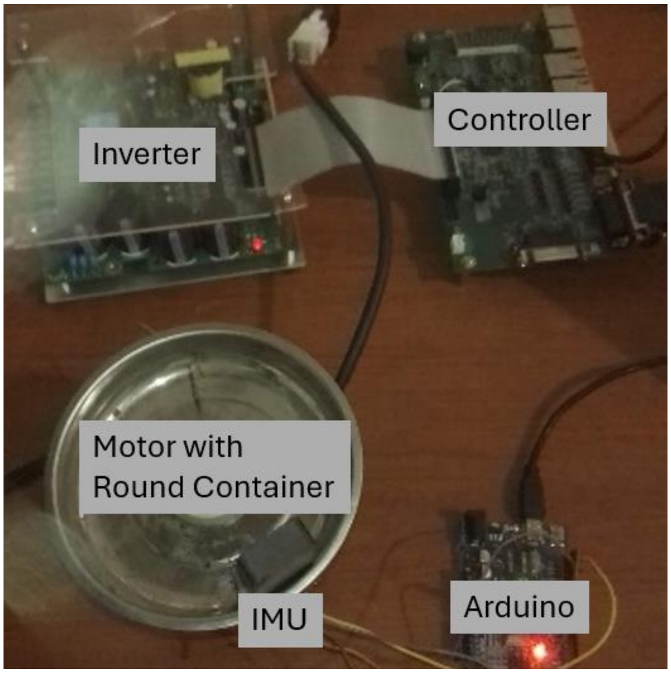


Figure 25. AC Motor Experiment Setup

This model is a surrogate and simplified version of a vehicle with an engine that turns a wheel. For example, the container connected to the end of the armature shaft is the wheel and the AC motor is the engine that rotates the wheel.

There are three sensors to monitor the AC motor dynamics. A camera to capture images of the simulated vehicle IP, a microphone to record the audio of the AC motor, and an IMU device to capture the vibration data of the AC motor. The IMU is attached to the side of the AC motor as experimented with.

3.13. Motor Faults Experiments and Data Collections

The simulated faults are introduced to the AC motor system by manipulating the round container connected to the end of the armature shaft.

The simulated faults examined are unbalanced wheel fault, wheel bearing fault, wheel load fault, and wheel mounting fault. The rotation motion and effects from faults and disturbances are collected using the sensors and provided to AI models for training.

To simulate an unbalanced wheel fault, a magnetic block object weighing approximately 0.24 lbs. and a 20mm x 50.5mm dimension is attached to the round container. This is to simulate incorrect weights on the vehicle wheel and collect the effects data.

The load fault is simulated by installing a device that will provide air resistance. The device has a fan propeller shape but it is installed perpendicular to the motor so that it would provide the maximum air resistance. For example, the motor would need to push through the air resistance brought on by this device as it rotates.

The wheel bearing fault simulation is executed with 35 round marbles that weigh a total of 0.04 lbs. and a radius of 7mm. The disturbance is applied to the rotating motor and as the articles collide in the round container, this would simulate symptoms of a bearing fault. As in bearing fault, broken particles collide as the wheel rotates and cause vibration and noise.

The wheel mounting fault is a scenario where the lug nut of the wheel is not properly mounted. This is simulated by loosening the mount connecting the round container and the armature shaft of the AC motor. The mounting issue will cause a wobbling effect on the AC motor and simulate wheel wobbling on a vehicle's wheel in motion.

The instantiated four rotational faults show dynamic patterns and features that are different from one another. The repetitive patterns show the disturbance as the motor rotates and reaches the fault condition. The vibration amplitude waveforms are recorded as channels 1, 2, and 3 as axes X, Y, and Z.

Below is the motor test bench for data collection of the wheel faults. The sound and vibration data are collected and used as training for AI tools.

Figure 26. Motor Testbench Setup

3.14. Vehicle Wheel Vibration Data

The vibration data from the vehicle wheel is collected using IMU attached to the top of the strut mount.

The hardware applied to collect the vibration data is shown below.

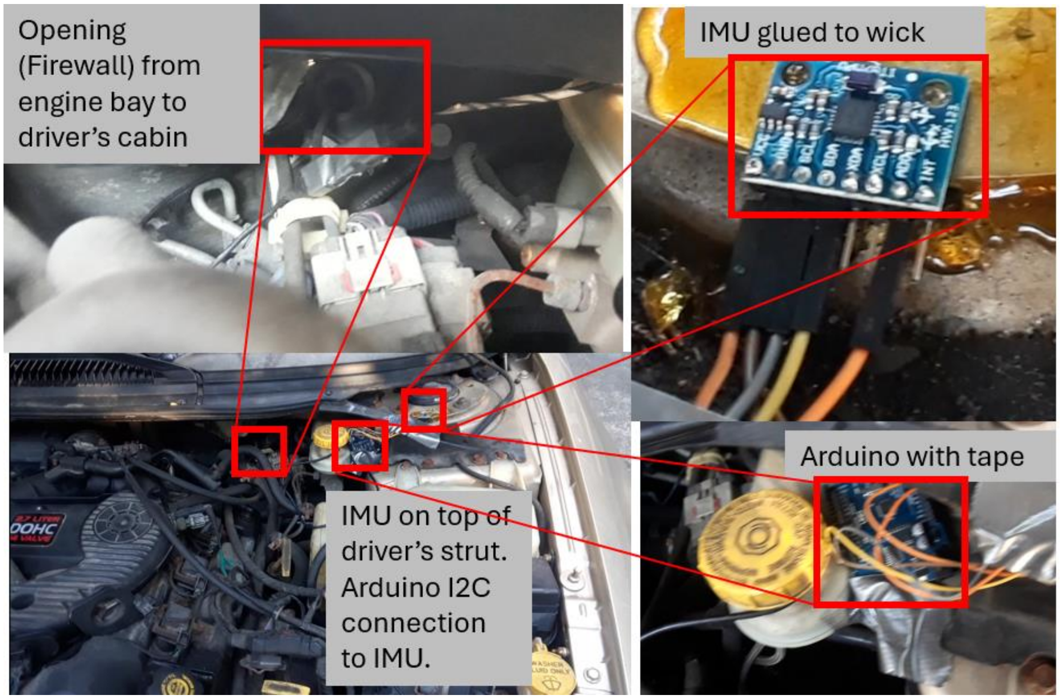


Figure 27. Vehicle Wheel Vibration Collection in Engine Bay

The Arduino connected to IMU receives the vibration data on top of the driver's side wheel and passes it to the laptop in the cabin. The USB cable is connected through the firewall opening from the engine bay to the cabin.

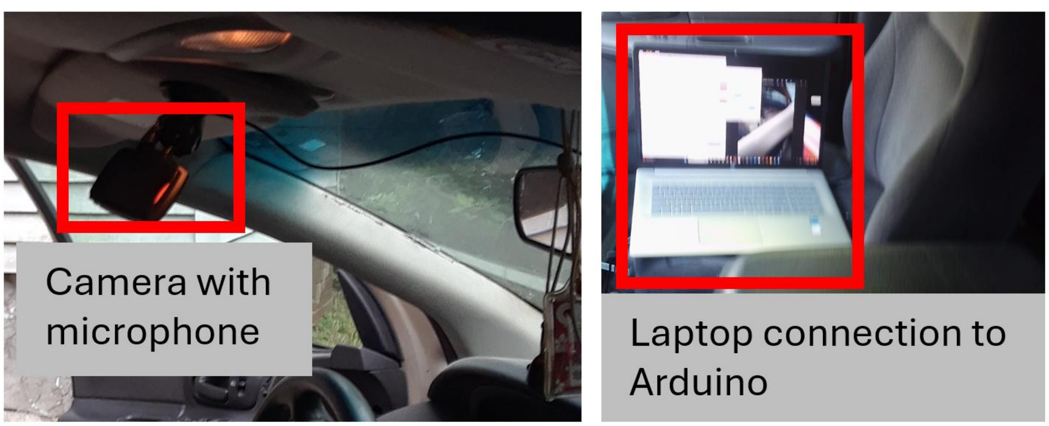


Figure 28. Vehicle Wheel Vibration Collection in Cabin

The camera with microphone records the Instrument Panel (IP) gauges and sound during the test. The laptop is connected to the Arduino and receives the vibration data on the wheel.

3.15. Inference of the Vehicle Wheel Faults

The simulation of fault effects on the AC motor with four different categories of faults is executed with the MATLAB tool. The data recorded from IMU connected to the AC motor is used with equations described in this section, to calculate the inferred vibrations and audio decibel data for the simulated vehicle wheel. For example, the equations will provide the vibration ratio value from the AC motor to the vehicle wheel. Similar math to process the audio recorded from motor operation with faults. The vibration magnitude values are also entered into math equations and calculations of effects on the vehicle speed from vehicle RPM. The higher vibration magnitude values will hinder the vehicle acceleration performance and show low speed on the speedometer.

3.15.1. Inference equations for vehicle wheel vibrations

The vibration magnitude data on time series are collected from the AC motor and reflect the values that would be from the vehicle. This way, we could estimate the vehicle wheel vibration magnitude values affected by the four faults while rotating. This is to assume a simple simulation where external factors such as wind, heat, ground friction, slippage, and others are limited.

First, assume below the vibration equation of the motor is a damped system without the faults disturbance.

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{9}$$

where:

m is the mass of the system.

$\ddot{x}$ is the acceleration.

c is the damping coefficient.

$\dot{x}$ is the velocity.

k is the stiffness of the system.

x is the displacement from the position.

Using the equation and damped natural frequency and the damping ratio is expressed as,

$$\zeta = \frac{c}{2\sqrt{mk}} \tag{10}$$

$$\omega_d = \omega_n \sqrt{1-\zeta^2} \tag{11}$$

$$\omega_{ns} = \frac{2\pi N_s}{60} \tag{12}$$

$$\omega_{nl} = \frac{v}{r} \tag{13}$$

$$N_s = \frac{120 \cdot f}{P} \tag{14}$$

$$J_s = \frac{1}{2} m_s r_s^2 \tag{15}$$

Where:

ζ is damping ratio

ω_d is damped natural frequency

ω_{ns} is damped-less natural frequency for motor

ω_{nl} is damped-less natural frequency for wheel

N_s is damped-less motor synchronous speed (rpm)

k_s is equivalent stiffness in motor (Nm/rad)

J_s is moment of inertia of the rotor (kg m^2)

m_s is mass of rotor (kg)

r_s is radius of motor (m)

As forced disturbance and wheel faults are introduced to the motor becomes:

$$m\ddot{x} + c\dot{x} + kx = F_0 \cos(\omega t) \tag{16}$$

where:

F_0 is the vibration amplitude of the introduced imbalance force,

ω is the angular velocity of a rotating motor.

The assumption of our motor as a damped harmonic oscillator as an external force of four fault cases is applied to equation. The dynamic response from the faults will be approximated.

The steady-state solution with sinusoidal external force would give us the vibration amplitude V equation as:

$$V = \frac{\frac{F_0}{m}}{\sqrt{(\omega_n^2 - \omega^2)^2 + (2\zeta\omega_n\omega)^2}} \tag{17}$$

Find the vibration amplitude of the vehicle wheel with the parameters of m_l, k_l, c_l, ω_l and motor wheel with parameters of m_s, k_s, c_s, ω_s by using amplitudes expressed as:

$$\frac{V_s}{V_l} = \frac{(\frac{F_0}{m_s})}{(\frac{F_0}{m_l})} \times \frac{\sqrt{(\omega_{nl}^2 - \omega_l^2)^2 + (2\zeta_l \omega_{nl} \omega_l)^2}}{\sqrt{(\omega_{ns}^2 - \omega_s^2)^2 + (2\zeta_s \omega_{ns} \omega_s)^2}} \tag{18}$$

For this case, we will assume that external disturbance force F_0 is proportional to the mass of the object. The vehicle wheel and motor would have different effects from the external force due to its difference in mass. The above can be simplified to below using dimension geometrical scaling and mass volume relationships in this vibration system as $m \propto \mathrm{D}$:

$$\frac{V_s}{V_l} \approx \frac{D_l^3}{D_s^3} \times \frac{\sqrt{(\omega_{nl}^2 - \omega_l^2)^2 + (2\zeta_l \omega_{nl} \omega_l)^2}}{\sqrt{(\omega_{ns}^2 - \omega_s^2)^2 + (2\zeta_s \omega_{ns} \omega_s)^2}} \tag{19}$$

Below is the vibration frequency ratio equation with damping and stiffness effects of the motor:

$$\frac{f_s}{f_l} \approx \frac{D_l}{D_s} \times \frac{\omega_{ds}}{\omega_{dl}} \tag{20}$$

where and is the damped natural frequency of the motor and vehicle wheel.

Assume the vehicle tire diameter is 0.4m (16”) weighing 45.28 lbs (wheel and tire) and the round container attached to the motor has 0.11 m (110 mm) diameter with a weight of 0.36 lbs.

To approximate stiffness of metal container k_s, Young’s modulus (E) is used as

$$k_s = \frac{E \cdot A}{L}, \tag{21}$$

where,

k_s = spring constant (N/m)

E = 200 GPa. This is Young's modulus of the steel material (N/m²)

A = $0.03m \cdot 0.002m = 0.00006m$ (width 30mm and thickness 2mm). This is cross-sectional area (m²) through which the force is applied.

L = 0.11m. Length or effective length over which the deformation occurs (m)

$$k_s = \frac{E \cdot A}{L} = \frac{(200 \times 10^9 N / m^2) \cdot (0.00006m^2)}{0.11m}$$

$$k_s = \frac{200 \times 10^4 N}{0.11}$$

$$k_s \approx 18MN / m$$

This means the effective spring constant of the metal container is approximately 18 M N/m. The same approach with steel E value is used approximately for 16” steel vehicle wheels with metal rims and gets approximately 1966.48 M N/m. The wheel has an inner diameter of 0.5m, a rim width of 0.2m, and a thickness of 0,01m. The damping ratio of sedan cars is 0.3 as stated, this value is used for vehicle wheels and 0.07 for induction motor frames for simplicity.

To calculate the non-damp angular velocity for the motor, the value of the reported stiffness of 0.9×106 in-lb/rad (98.7 kNm/rad) is applied and the calculated values are applied to below.

$$D_s = 0.11m, D_l = 0.40m$$

$$m_s = 0.16kg, m_l = 20.53kg$$

$$k_s = 18.18MN / m, k_l = 1966.48MN / m$$

$$\zeta_s = 0.07, \zeta_l = 0.3$$

Using a frequency of 60 Hz and 4 poles in the motor:

$$N_s = \frac{120 \cdot f}{P} = \frac{120 \cdot 60}{4} = 1800rpm$$

$$\omega_{ns} = \frac{2\pi N_s}{60} = \frac{2\pi \cdot 1800}{60} \approx 188.5 rad / s$$

For simplicity with a wheel diameter of 16inch and vehicle driving state of 1500 rpm, and a velocity of 42 mph (18.77 m/s) will not include external factors such as road conditions, frictions, tire conditions, etc.

$$r = \frac{16inches}{2} \times 0.0254 \approx 0.20m$$

$$\omega_{nl} = \frac{v}{r} = \frac{20m / s}{0.20m} \approx 98.5 rad / s$$

Using the calculated damp-less angular speed:

$$\omega_{ds} = \omega_{ns}\sqrt{1-\zeta_s{}^2} = 188.5\sqrt{1-0.07^2}$$

$$\approx 188.03 rad / s$$

For vehicle wheel, we would get:

$$\omega_{dl} = \omega_{nl}\sqrt{1-\zeta_l{}^2} = 98.5\sqrt{1-0.3^2}$$

$$\approx 93.96 rad / s$$

So we would compute the vibration ratio as

$$\frac{V_s}{V_l} = \frac{0.40^3}{0.11^3} \times \ldots$$

$$\frac{\sqrt{(98.5^2 - \omega_l^2)^2 + (2 \times 0.3 \times 98.5 \times \omega_l)^2}}{\sqrt{(188.5^2 - \omega_s^2)^2 + (2 \times 0.07 \times 188.5 \times \omega_s)^2}}$$

We could approximate as,

$$\frac{f_s}{f_l} \approx 3.63 \times \frac{188.03}{93.96} \approx 7.26$$

The above indicates that the dimensional ratio of 3.63 and the disturbance vibration response ratio between motor and vehicle wheels is approximately 7.26. This means the external faults applied to the motor f_s is 7.26 times more than the vehicle wheel f_l. This ratio is used to infer the vibration fault magnitude applied to the vehicle wheel using the vibration response data. For example, the vibration recorded on the motor is inferred to be 7.26 larger due to having almost no load, smaller volume, and weight compared to the wheel installed on the vehicle.

3.16. Inference equations for vehicle speed

The speed of Revolutions Per Minute (RPM), is estimated in the simulated world, to speedometer reading on the IP using the following vehicle speed information.

$$Speed = \frac{RPM \times D \times \pi}{N} \times \frac{60}{1000} \tag{22}$$

where:

D is tire diameter in meters

N is the gear ratio (1 for direct drive)

Circumference is $D \times \pi$

Speed is in kilometers per hour (km/h)

A factor of 60 is used to provide hours from minutes

A factor of 1000 is used to provide kilometers from meters

Distance per minute: $RPM \times D \times \pi$

Wheel RPM is

$$\frac{Engine\ RPM}{Gear\ Ratio \times Final\ Drive\ Ratio} \tag{23}$$

In our simulation, we have a tire diameter of 16 inches which is 0.4 meters and if the engine is turning at 1000 RPM, then we would have a speed of 46.83 mph as shown below:

Distance per minute = Wheel RPM $\times$ Circumference

$$\text{Speed(mph)}=\frac{\text{Distance per minute} \times 60 \times 0.6214}{1000}$$

$$\text{Speed(mph)}=\frac{1000\times 0.4\times \pi \times 60}{1000}\times 0.6214$$

$$\text{Speed(mph)}=46.83$$

The calculated speed of the vehicle is simulated on the instrument panel RPM gauge and Speed odometer as shown below.

Figure 29. Instrument Panel in MATLAB

The IP on the left of the screen is simulated with four circular gauge blocks in Simulink of fuel, RPM, speed, and temperature. The RPM gauge show is inferred from motor RPM and the speedometer gauge value is computed.

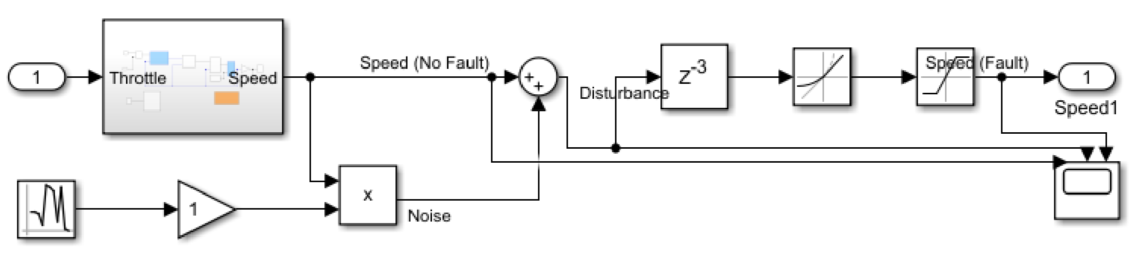


Figure 30. MATLAB model algorithm for vehicle speed

Above is the MATLAB tool that calculates the vehicle throttle value into vehicle speed taking into account of gear shifts. The values have been calibrated to suit our needs and our experiment conditions. The calculated ratio between the motor and wheel has been considered in the calculations.

Below is one of the block parameters that is used to calibrate the magnitude of the disturbance force applied to the vehicle wheel which inhibits the speed of the vehicle. The higher the gain value will mean more disturbance and less capability for the vehicle to smoothly speed up.

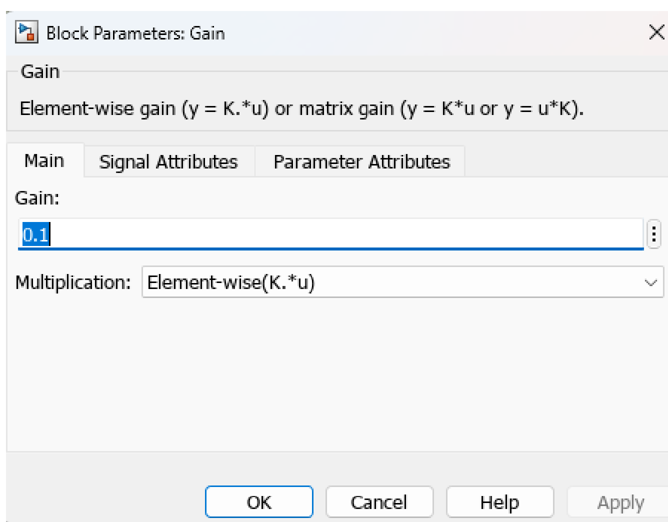


Figure 31. Simulink block to simulate wheel faults

The response to the fault is calibrated using the value of Gain value. The block parameters are shown below. The parameter values are calibrated to show the effects of disturbance force and wheel faults on the vehicle speed on the speed gauge inside the instrument panel.

Table 7. Simulink Wheel Fault Severity Setting

Wheel Fault Type	Description	Severity (1-4) 4: Most severe.	Parameter values (Gain, Slope, Saturation)
Bearing Fault	Damaged bearing. Noisy with low pitch sound of particles roaming. Wheel turns with slight vibration and shake. Hinders vehicle speed slightly.	3 (Moderate severity)	2,50,2000
Load Fault	Vehicle is not balanced and weight load is one sided. The tire on the wheel could show uneven wear. Almost no damage.	1 (Not Severe)	0.1, 70, 2000
Mount Fault	Vehicle wheel is not mounted properly and could be missing lugnuts. Vehicle wheel is about to fall out. Heavy vibration and noise. Vehicle can not speed up more than 5 mph..	4 (Extreme Severity)	2,40,200
Unbalanced Fault	The weights on the wheel is missing or wrong weights are added. The wheel shows uneven wear and vibrates that could hinder vehicle speed slightly.	2 (Slight Severe)	1,60,2000

3.16.1. Equations for vehicle sound

The decibel level of a vibration signal is calculated using the formula

$dB_{vibration} = 20 \times \log_{10}(\frac{RMS_{vibration}}{RMS_0})$, where $RMS_{vibration}$ is the root mean square of the measured vibration and RMS_0 is a reference RMS value. This approach is discussed in detail in sources on vibration analysis and acoustics. Using the formula, the recorded motor audio data is adjusted to audio that would be coming from the vehicle in the simulation.

The actual sound pressure level is calculated using MATLAB tool after collection from vehicle wheel and motor. The USB sound recording device is zipped and tied to the front of the driver's wheel and the microphone is pointed towards the inner side. The setup is shown below.

Figure 32. Wheel Sound Recording Setup

The sound is recorded when the vehicle is idling at 800 RPM and approximately at 1500 RPM driving at 40 MPH. The timing of the recorder and vehicle RPM and speed are monitored using the camera in the cabin. Before each test, the recorder and camera are time-synced so that the speed of the vehicle matches the sound recorded.

Below is the same concept with the vehicle but with the motor on the test bench. The same USB sound recorder is set next to the rotating motor and the microphone is pointed towards the metal container. The test setup is shown below.

Figure 33. Motor Testbench Setup

3.16.2. Inference equations for fault decision making

There are four faults in the simulation. Suppose we will label the unbalanced wheel fault as “A”, the wheel bearing fault as “B”, the wheel load fault as “C”, and the wheel mounting fault as “D”. Each sensor device will be labeled as well. For example, the Camera would be c, the Microphone is m, and IMU is i. So if the value is from

Camera on fault "A" then the label is Ac. This is expressed as follows, Ac, Bc, Cc,…∈ [0,1] ⊂ $\mathcal{R}$. Similarly, Am, Bm, Cm ∈,… [0,1] ⊂ $\mathcal{R}$ and Ai, Bi, Ci,…∈ [0,1] ⊂ $\mathcal{R}$. The classification of the devices and faults is shown below.

Table 8. Devices and Faults Classification for Wheel Faults

Devices	"A" Wheel Unbalanced Fault	"B" Wheel Bearing Fault	"C" Wheel Load Fault	"D" Wheel Mounting Fault
Camera	A_c	B_c	C_c	D_c
Microphone	A_m	B_m	C_m	D_m
IMU	A_i	B_i	C_i	D_i

The fault-predicted values from devices shown are processed by Sugeno FIS with the equations below.

$$L_A = \frac{\sum_{k=1}^{27}(As_k * Aw_k)}{\sum_{k=1}^{27} As_k}$$

$$L_B = \frac{\sum_{k=1}^{27}(Bs_k * Bw_k)}{\sum_{k=1}^{27} Bs_k}$$

$$L_C = \frac{\sum_{k=1}^{27}(Cs_k * Cw_k)}{\sum_{k=1}^{27} Cs_k}$$

$$L_D = \frac{\sum_{k=1}^{27}(Ds_k * Dw_k)}{\sum_{k=1}^{27} Ds_k}$$

Where:

L_A is the likelihood value that fault "A" is occurring.

As_k is the firing strength of fault "A"

Aw_k is the weight of fault "A"

…

L_D is the likelihood value that fault "D" is occurring.

Ds_k is the firing strength of fault "D"

Dw_k is the weight of fault "D"

Equations are computed with each singleton membership function and the highest likelihood value of the fault is the predicted wheel fault.

CHAPTER FOUR

RESULTS

4.1. Overview of Experimental Setup

4.1.1. “A” Faulty Battery

The vehicle is installed with a depleted battery with a voltage of less than 12V. This battery is defective and cannot support the required 1.3kW-hours to start a sedan. The vehicle will emit fast-clicking noise and show blinking lights on the instrument panel or if damaged then there will only be a single click noise.

The goal of this experiment case is to define and record the unique features and traits of vehicles not starting due to having a bad battery. This would eliminate not starting due to a bad battery and could continue with a bad starter, bad gas, bad ignition, and other causes that would have other tell-tale signs.

4.1.2. “E” Faulty Wheel

The vehicle wheel is added with extra weights and will cause the wheel to be unbalanced. This vehicle will now have an unbalanced wheel and produce more vibrations and noise when driven due to it being wobbly. The effects of an unbalanced wheel are recorded with the IMU installed on top of the strut.

The goal of this experiment case is to define and record the unique features and traits of vehicles having noise and shaking motion when driven. There could be several reasons such as a bad wheel, bad lower control arm, bad stabilizer, bad axel, and so forth. Each defect would emit a unique vibration and noise and can be classified.

4.2. Training and Validation Results

"A" Faulty Battery Case: The recorded visual, audio, and 3-axis data are recorded and processed through trained VisNet, AudNet, and VibNet with the vehicle having a bad battery. The number of predictions is shown below on the fault classifications of Faulty Battery, Faulty Alternator, Faulty Starter, Faulty Spark Plug, Faulty Wheel, and Faulty Engine. The Faulty Battery had the highest recognition of 89 using the camera. Microphone recordings processed through Yam-net showed the highest value of 79. IMU had the highest value of 22 as Faulty Spark Plug. The results displayed of Bad Battery and Bad Wheel detection with a combination of VisNet, AudNet, and VibNet with decision-making using fuzzy logic resulted in Faulty Battery.

Devices	"A" Faulty Battery	"B" Faulty Alternator	"C" Faulty Starter	"D" Faulty Spark Plug	"E" Faulty Wheel	"F" Faulty Engine (Water Pump, EGR)
Camera	87	4	1	2	5	1
Microphone	79	5	1	5	6	4
IMU	3	4	66	14	7	6

Figure 34. Validation Results on Faulty Battery

"E" Faulty Wheel Case: The recorded visual, audio, and 3-axis data are recorded and processed through trained VisNet, AudNet, and VibNet while the vehicle is in motion with the unbalanced wheel. The number of predictions is shown below on the fault classifications of Faulty Battery, Faulty Alternator, Faulty Starter, Faulty Spark Plug, Faulty Wheel, and Faulty Engine. The Faulty Wheel had the highest recognition of

35 using the camera. Microphone recordings processed through AudNet showed the highest value of 33 for Faulty Wheel. IMU had the highest value of 66 as a Faulty Wheel. The displayed results of Bad Wheel detection with a combination of VisNet, AudNet, and VisNet with decision-making using fuzzy logic resulted in a Faulty Wheel.

Devices	"A" Faulty Battery	"B" Faulty Alternator	"C" Faulty Starter	"D" Faulty Spark Plug	"E" Faulty Wheel	"F" Faulty Engine (Water Pump, EGR)
Camera	11	12	12	12	43	10
Microphone	14	10	12	7	45	12
IMU	2	7	5	35	45	6

Figure 35. Validation Results on Faulty Wheel

4.3. IMU Vibration Results

The vibration data from IMU installed in the vehicle engine bay is recorded and graphed as below.

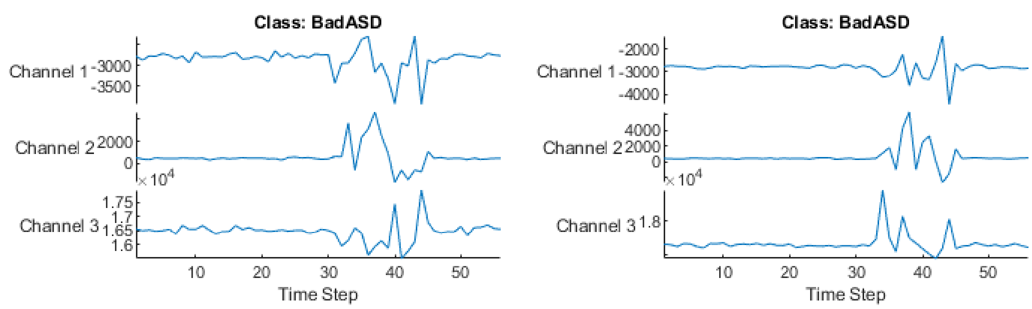


Figure 36. Bad ASD (Relay) Fault Vibrations

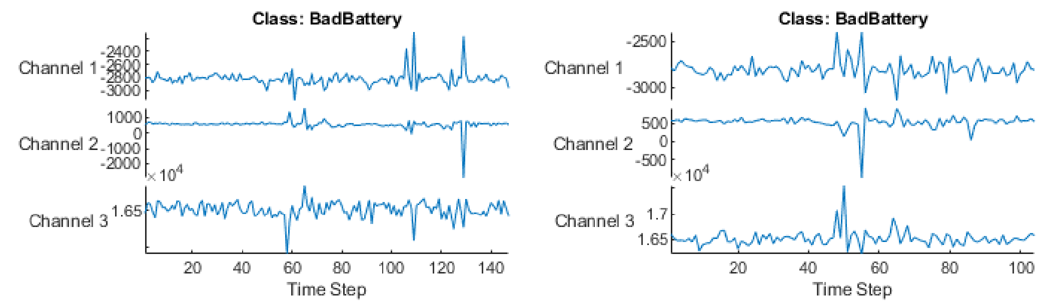


Figure 37. Bad Battery Fault Vibrations

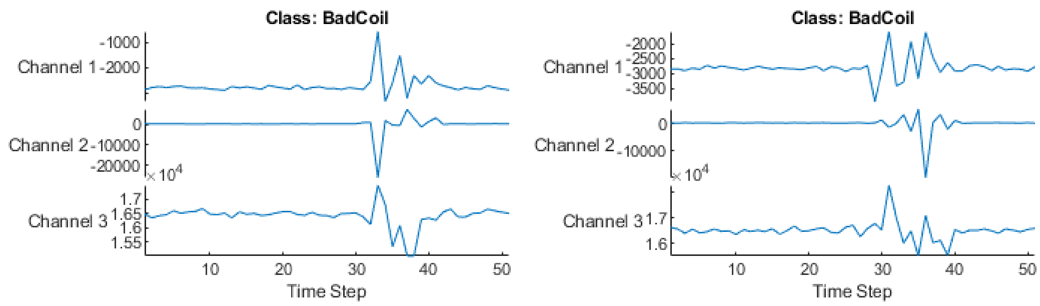


Figure 38. Bad Coil Fault Vibrations

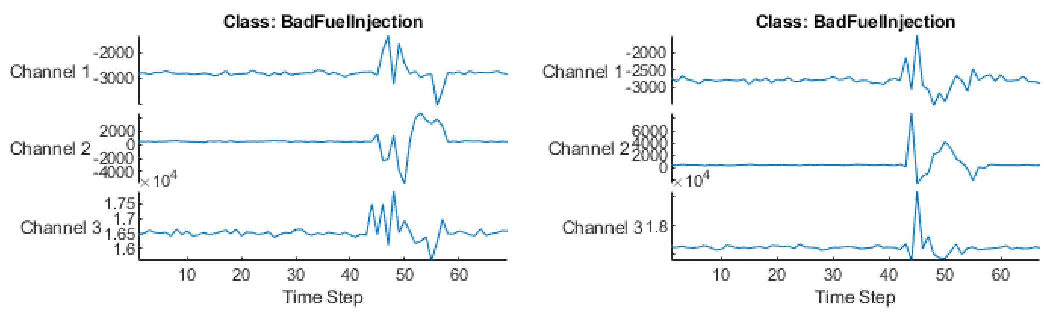


Figure 39. Bad Fuel Injection Fault Vibrations

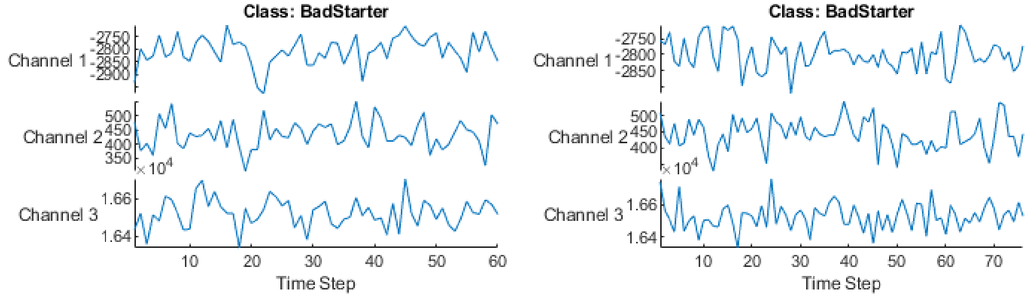


Figure 40. Bad Starter Fault Vibrations

Channel 1, 2 & 3 in each graph shows the x-axis, y-axis, and z-axis acceleration data when trying to start the vehicle with fault. As shown in the graph, there are similarities but also distinctions between each fault scenario. The differences could be seen from the frequency of the axis vibration and also the magnitude of the vibration. This vibration data helps complement the similar audio data recorded that is almost indistinguishable from human hearing.

Below are graphs for vehicle wheel faults. Channel 1 is x-axis data, channel 2 is y-axis data, and channel 3 is z-axis data. The data is collected with a time step of milliseconds.

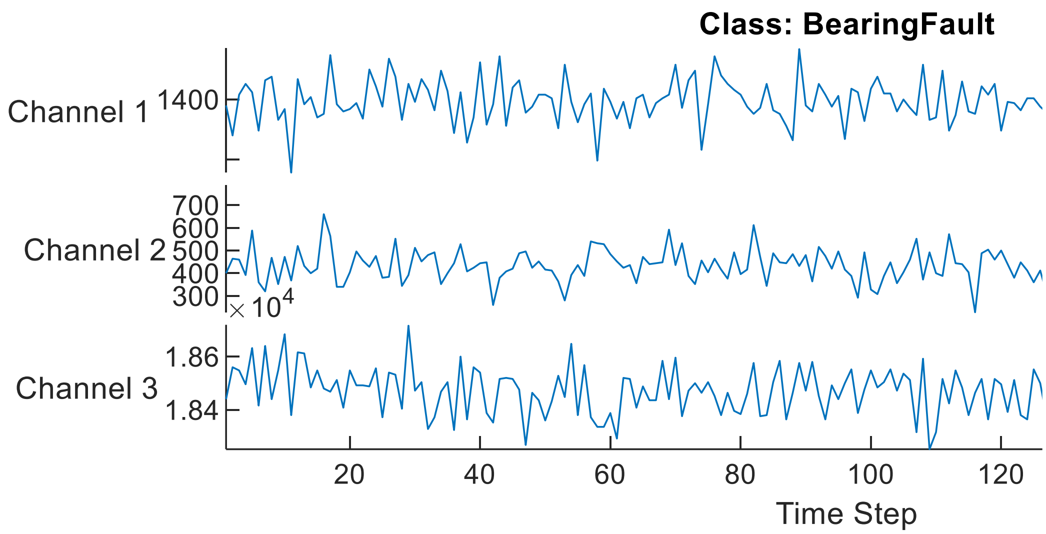


Figure 41. Vehicle Wheel Bearing Fault Vibration

The bearing fault vibration would indicate the frequency and magnitude of small marbles and objects inside rotating with the motor. This is the scenario where the wheel bearing is damaged and the damaged pieces of the ball bearing are rotating inside the wheel causing forced disturbance and vibrations.

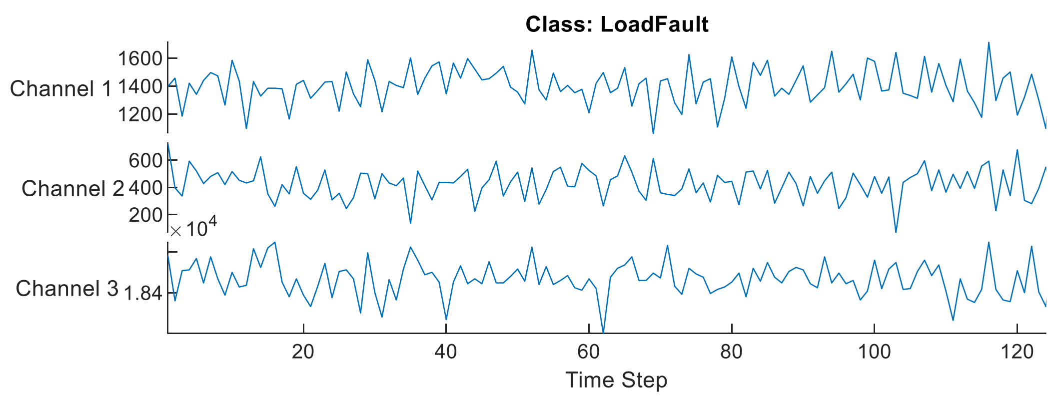


Figure 42. Vehicle Wheel Load Fault Vibration

The load fault vibration is a fault caused by extra load on each turn of the rotation. This is the scenario where the vehicle is not load balanced and one side of the vehicle is heavier than the other causing the wheel to not able to rotate properly. A portion of the turn will cause forced disturbance and additional load and vibration magnitude value.

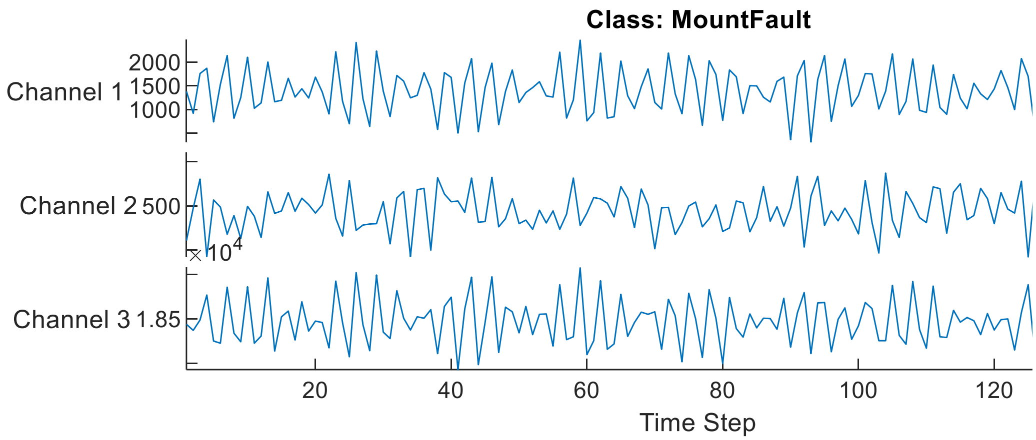


Figure 43. Vehicle Wheel Fault Vibration

The wheel mount fault vibration is similar to the previous load fault but because the wheel mount is not properly installed and attached to the motor, there are higher frequency of disturbance as seen in the above graph. The wheel is not mounted properly which causes free movement in all x, y, and z axis and causes unwanted vibration. So, this is sound in the graph as a high frequency of vibrations.

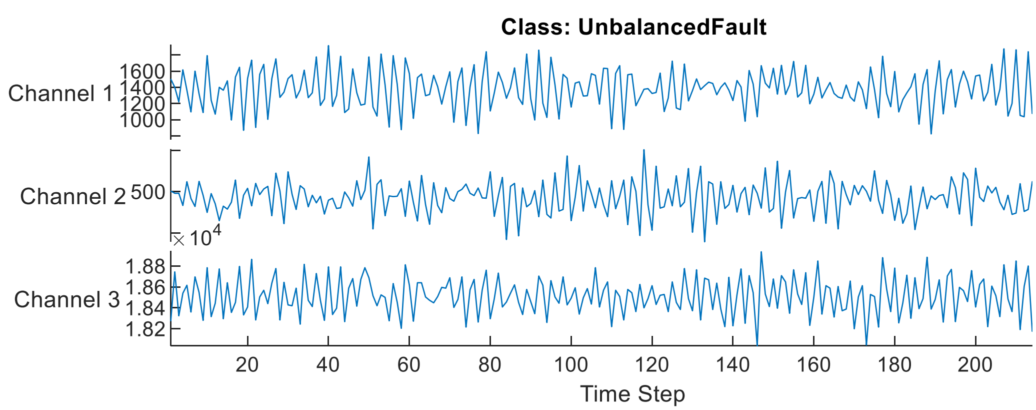


Figure 44. Vehicle Wheel Unbalanced Fault Vibration

The unbalanced fault is created by adding an extra block of weight to the wheel. In vehicle wheels, this would be weights that are added to the wheel to help smooth the rotation. But to create a bad wheel with an unbalanced fault, an additional block of weight is applied to the system and adds force disturbance at each turn. This shows on the graph a high frequency of vibrations as the wheel turns as it is now unbalanced.

4.4. Vehicle vibration data in idling and driving

The motor is run without external force or faults and vibration data is collected rotating at approximately 800 rpm. To filter out the extreme and usually invalid values, the accumulated values maximum and minimum are the second largest and smallest. The collected vibration magnitude data are shown below.

Table 9. Motor Vibration Data

Motor State	Vibration	X axis	Y axis	Z axis
Not running (0 rpm)	Min	-88	-84	-220
	Max	260	268	320
	Displacement	348	352	540
Running (~800 rpm)	Min	-520	-1092	-504
	Max	852	1204	412
	Displacement	1372	2296	916
Running (~1500 rpm)	Min	-11480	-24024	-17237
	Max	12124	21188	6464
	Displacement	23604	45212	23701

Below is a graph of the motor vibration displacements recorded with rpm of vehicle states of no start, idling, and driving.

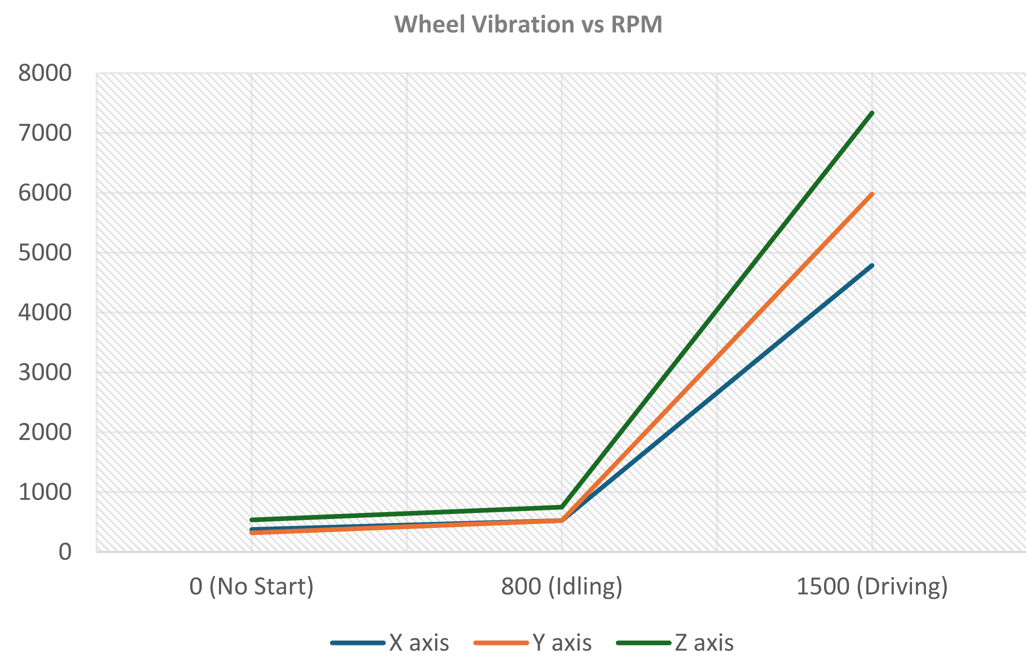


Figure 45. Wheel Vibration vs RPM

Below is vibration magnitude data collected when the vehicle engine is not running, idling at approximately 800 rpm, and driving at approximately 1,500 rpm.

Table 10. Vehicle Vibration Magnitude Values

Vehicle State	Vibration	X axis	Y axis	Z axis
Not started (0 rpm)	Min	-112	-120	-172
	Max	264	200	364
	Displacement	376	320	536
Idling (~800 rpm)	Min	-220	-236	-400
	Max	304	288	348
	Displacement	524	524	748
Driving (~1500 rpm)	Min	-3216	-3424	-3896
	Max	1572	2556	3440
	Displacement	4788	5980	7336

Below is a graph with recorded wheel vibration displacement data. The data are captured when the vehicle is not started, idling at 800 rpm, and driving mode at 1,500 rpm.

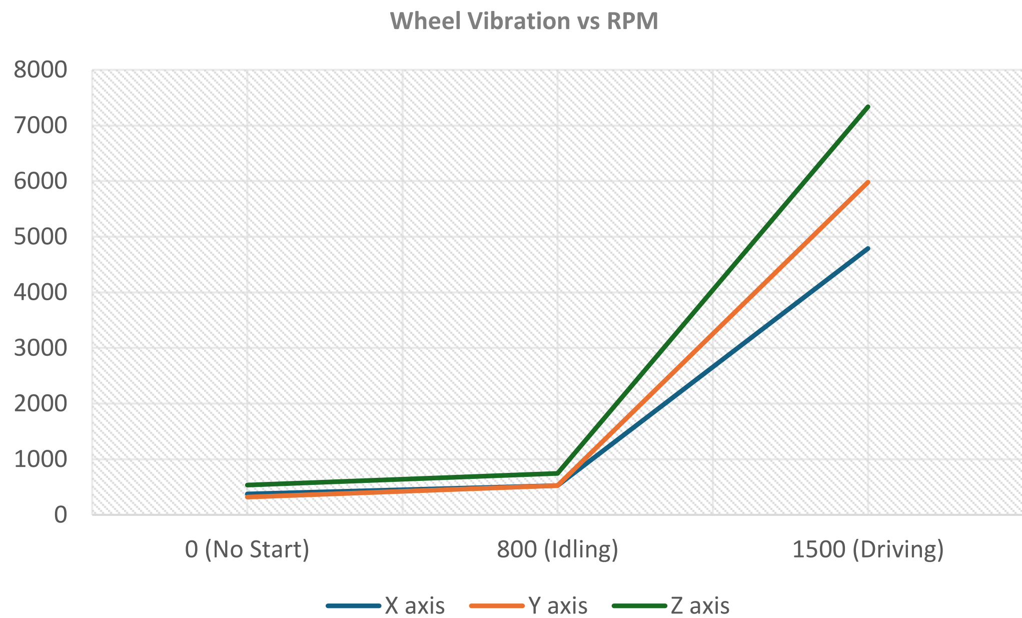


Figure 46. Wheel Vibration vs RPM

The Vibration Magnitude Ratio (VMR) is computed with $VMR = \frac{VA_M}{VA_W}$ where, VA_M is vibration amplitude from the motor and VA_W is vibration amplitude from the vehicle wheel.

Comparing the vibration magnitude from the motor to the vehicle wheel, the ratio is as follows:

Table 11. Vibration Magnitude Ratio (Wheel/Motor)

RPM	VMR	X axis	Y axis	Z axis
0 rpm	Motor	348	352	540
	Wheel	376	320	536
	VMR	0.92	1.1	1
~800 rpm	Motor	1372	2296	916
	Wheel	524	524	748
	VMR	2.61	4.38	1.22
~1500 rpm	Motor	23604	45212	23701
	Wheel	4788	5980	7336
	VMR	4.92	7.56	3.23

The ratio chart above shows similar results with a VMR value close to 1 when the motor and vehicle are not moving, which is expected. As rotation speed increases to 800 rpm, the vibration amplitude values for the motor are on average 2.73 times more than on the vehicle idling which is likely due to the bigger mass and heavier weight of gross vehicle weight 5,000 lbs. on the vehicle when heaviest. The weight of the motor is 1.75 lbs. which is incomparably light compared to the mounted vehicle wheel. Thus, vibration amplitude with a high rotational speed of 1,500 rpm and driving at approximately 40 mph shows less vibration. The motor is vibrating at a high amplitude average of 30,839 (digital reading) and VMR of roughly 5.23 times more than the wheel on the vehicle.

The vibration displacement ratio between the motor and wheel is shown in the graph below.

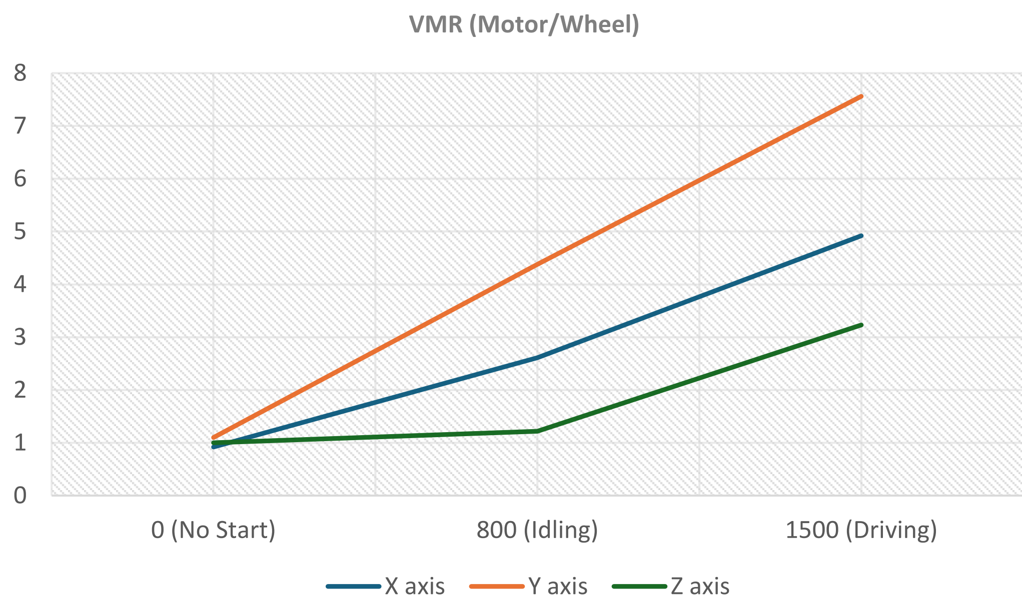


Figure 47. VMR (Motor/Wheel)

The approximate ratio between the motor and vehicle wheel is calculated with collected data when there is no external force or disturbance. The next section will discuss inference calculations applied to the surrogate model to infer the possible vibration impact on the vehicle wheel with rotation faults applied to the motor.

4.5. Results from Sound Pressure Comparison

4.5.1. Vehicle Idling

The sound pressure level is collected from the vehicle wheel and motor to assess the difference between the two. The sound is collected and then computed using MATLAB tool to visualize the difference between the two. Below is the sound pressure level. The wheel when the vehicle is idling, emits a higher peak level of 95 dB compared to 68 dB by the motor. The average difference between the two is similar to about 30 dB.

Table 12. Sound Pressure Level at 800 RPM

Sound Pressure Level	@800 RPM (idling)
Wheel	Peak@ 95 dB; Avg@ 82 dB
Motor	Peak@ 68 dB; Avg@ 54 dB

The visual representation of the sound level in time series and frequency is shown figure below. Below is the sound pressure level from the vehicle wheel with a peak of 95dB and an average of 82 dB.

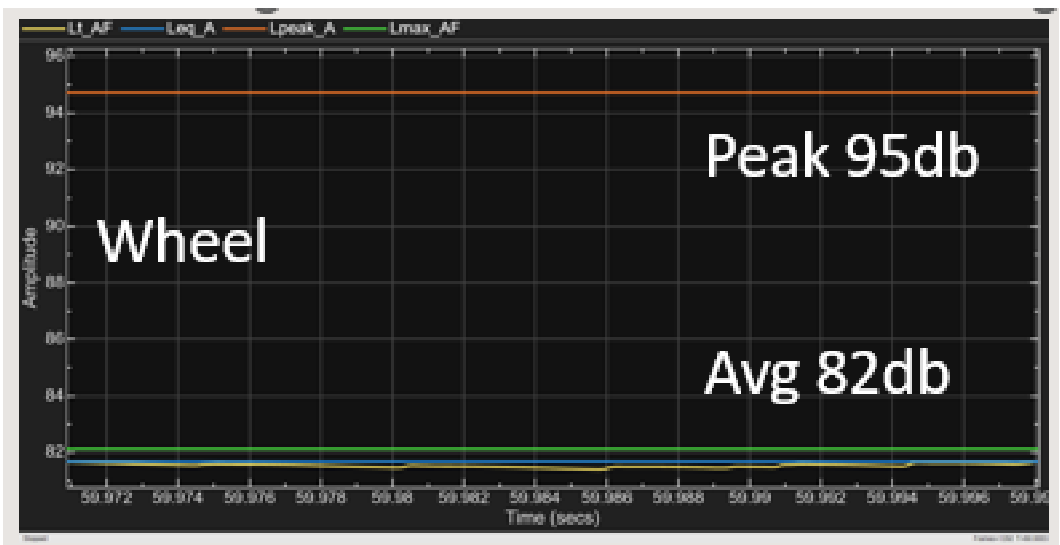


Figure 48. Wheel Sound Pressure Level

Below is the same condition of the rotating speed of 800 RPM on the motor. The sound pressure level peaks at 68 dB and averages at 54 dB.

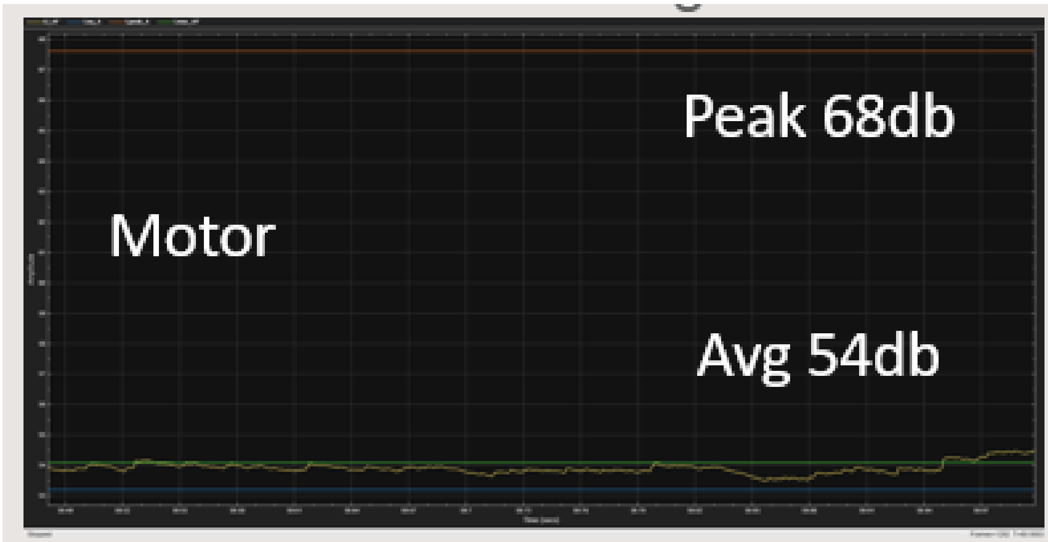


Figure 49. Motor Sound Pressure Level

The vehicle wheel sound emitted graphed with frequency is shown below. It shows that most of the high magnitude dB values are approximately 1 kHz.

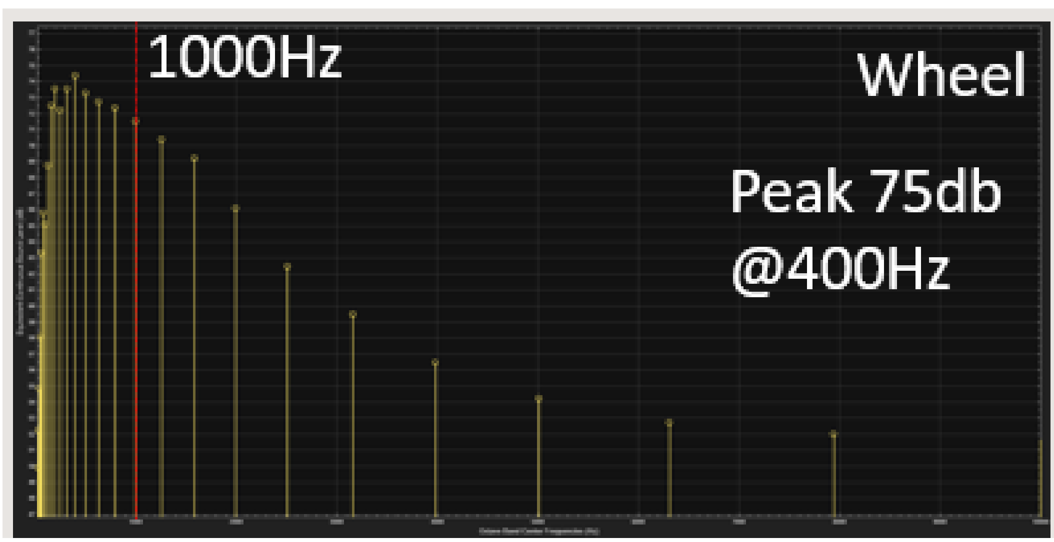


Figure 50. Wheel Sound Magnitude with Frequency

Comparing the above vehicle wheel frequency magnitude with the motor, we find that interestingly they are similar. The peak for the vehicle wheel is at approximately 400 Hz and the motor is at 800 Hz. This would mean that because the size of the motor is smaller than the vehicle wheel, the motor creates most sounds when rotating twice as fast as the motor inside the vehicle wheel. The graph of the motor is shown below.

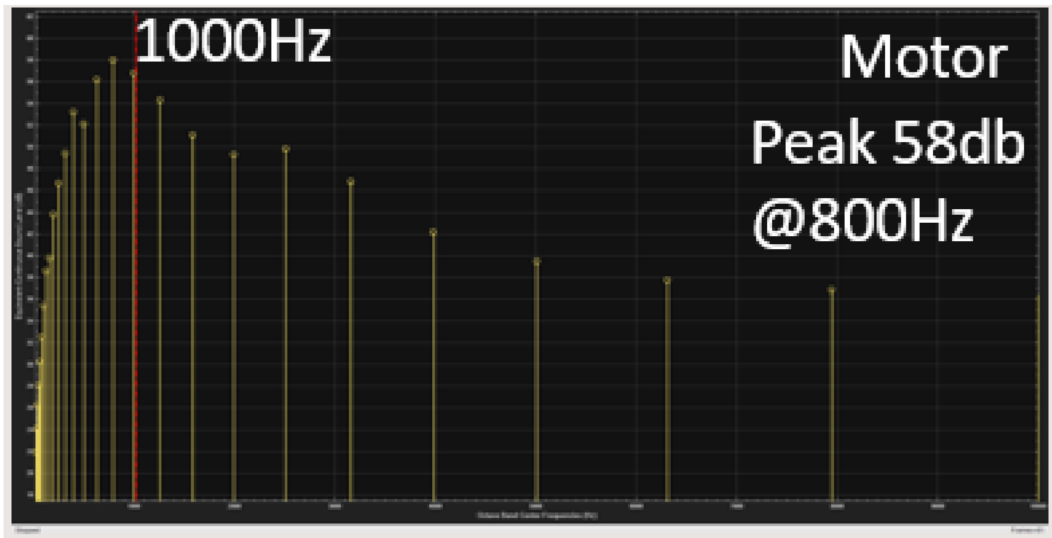


Figure 51. Motor Sound Magnitude with Frequency

4.5.2. Vehicle Traveling at 40 MPH

This time we will compare the sound pressure level on the wheel when the vehicle is traveling on public road at 40 MPH/ 1500 RPM. The collected data for vehicle wheels and motor is shown below.

Table 13. Sound Pressure Level at 1500 RPM

Sound Pressure Level	@1500 RPM (40 MPH)
Wheel	Peak@ 97 dB; Avg@ 85 dB
Motor	Peak@ 77 dB; Avg@ 63 dB

It is interesting to note that compared to the previous sound pressure comparison, we do not see a large difference in magnitude. The wheel shows only an increase of 2db when it rotates twice as fast from 800 RPM. This is similarly true for the motor as well with only a modest increase from a peak of 68 to 77 dB.

Below is a time series graph of the vehicle wheel when traveling at 40 MPH.

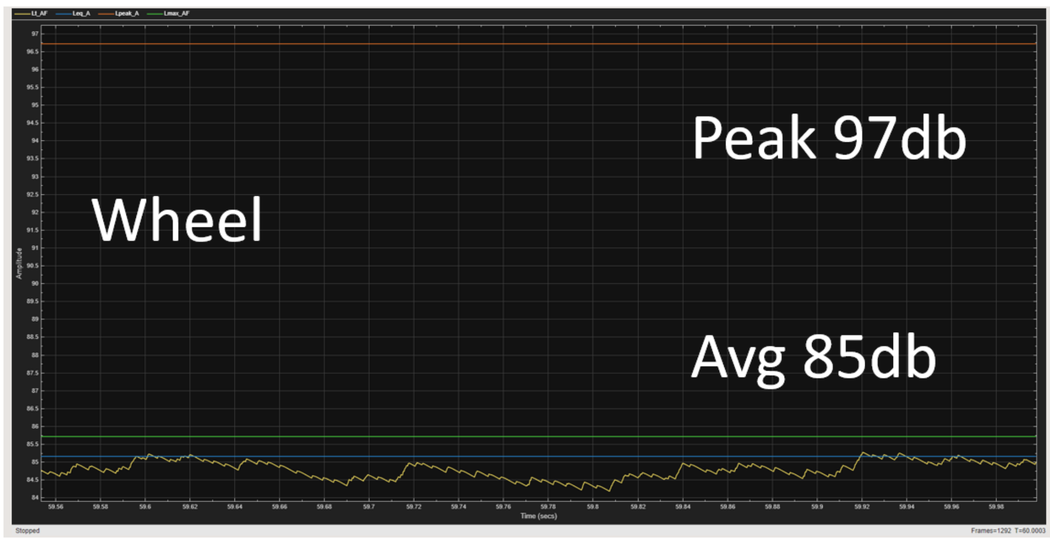


Figure 52. Vehicle Sound Magnitude at 40 MPH

Below is a same graph of the motor rotating at the same speed of 1500 RPM.

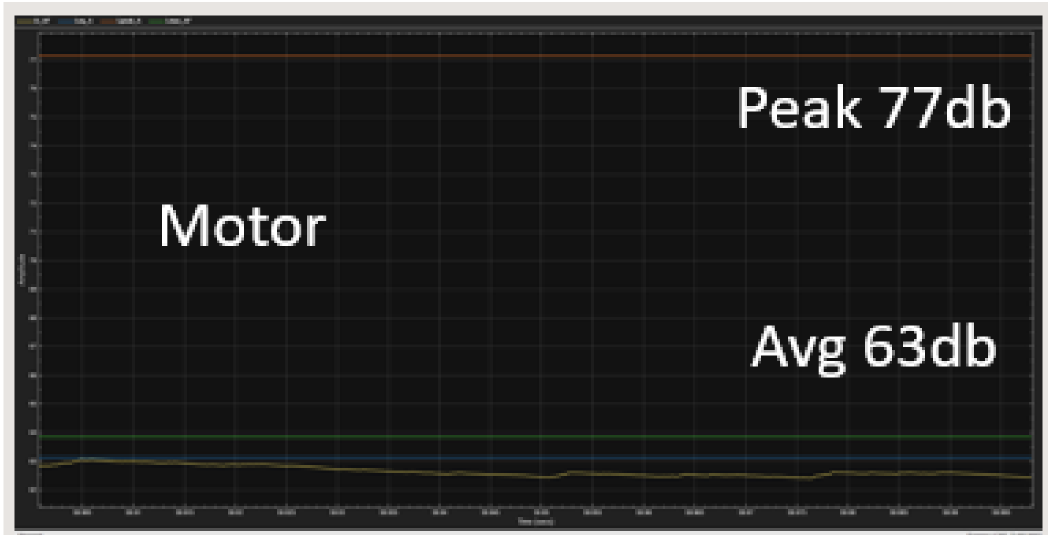


Figure 53. Motor Sound Magnitude at 1500RPM

The frequency graph below shows a peak magnitude of 79 dB emitting from the vehicle wheel in the range of 1.2 kHz.

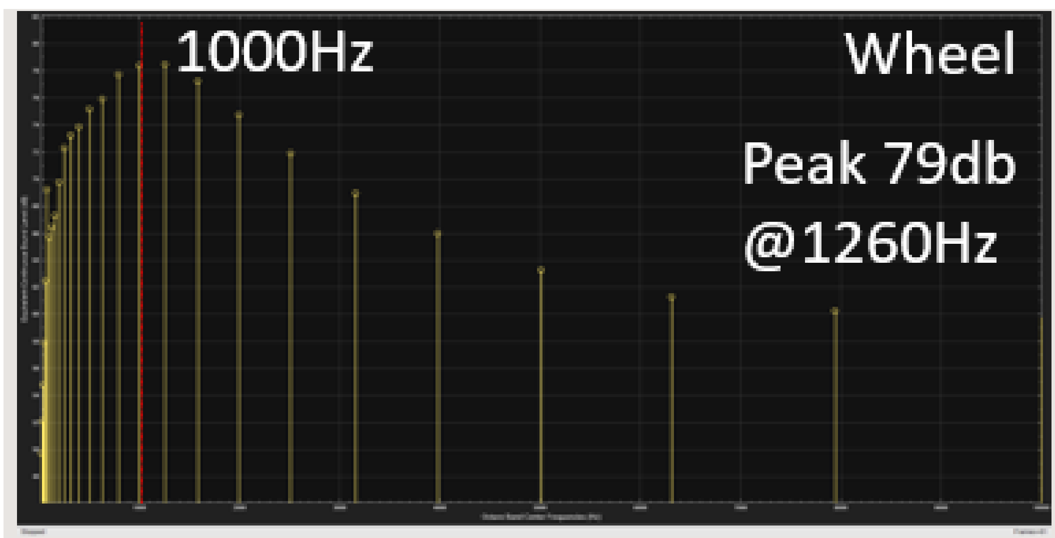


Figure 54. Vehicle Sound at 40 MPH with Frequency

The motor frequency graph displays the peak sound pressure level of 58 dB at 800 Hz. Compared to the previous frequency graph rotating at 800 RPM, the frequency magnitude values of the vehicle and motor are more similar. Below is a frequency graph for the motor rotating at 1500 RPM.

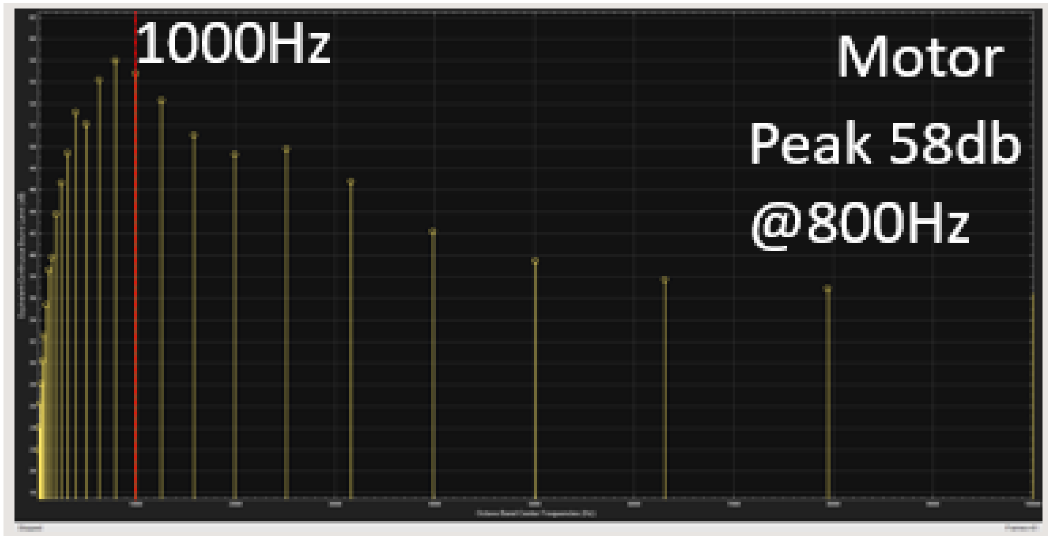


Figure 55. Frequency Graph of Motor at 1500 RPM

This is interesting as considering the obvious difference and power between the motor and vehicle, it is surprising that there isn't much noise coming from the vehicle idling and driving. Below are the comparison results in the table.

Table 14. Comparison at 800 RPM and 1500 RPM

Sound Pressure Level	@800 RPM (idling)	@1500 RPM (40 MPH)
Wheel	Peak@ 95 dB; Avg@ 82 dB	Peak@ 97 dB; Avg@ 85 dB
Motor	Peak@ 68 dB; Avg@ 54 dB	Peak@ 77 dB; Avg@ 63 dB

To summarize the comparison of sound pressure level between the vehicle wheel and motor rotating at 800 RPM and 1500 RPM, we would get the ratio result of about 1.3 as shown in the below table.

Table 15. Sound Pressure Level Ratio

Ratio	@800 RPM (idling)	@1500 RPM (40 MPH)
Avg Ratio (Wheel/ Motor)	1.4	1.3

4.6. Results from Forced Disturbance to Vehicle Speed

When there is no fault. The throttle and vehicle speed graph is as below.

Figure 56. Vehicle Speed Graph Legend

The graphs will use 3 data from the vehicle speed shown in the yellow line, the external disturbance force applied to the vehicle wheel in the form of wheel faults, and the vehicle speed shown in the red line in the graphs.

The below graph shows the normal slopes of vehicles when there is no external disturbance applied to the system. As the throttle value increases, the speed of the vehicle increases with it. The speed without fault rises higher and faster than the one displayed with a fault in the red line.

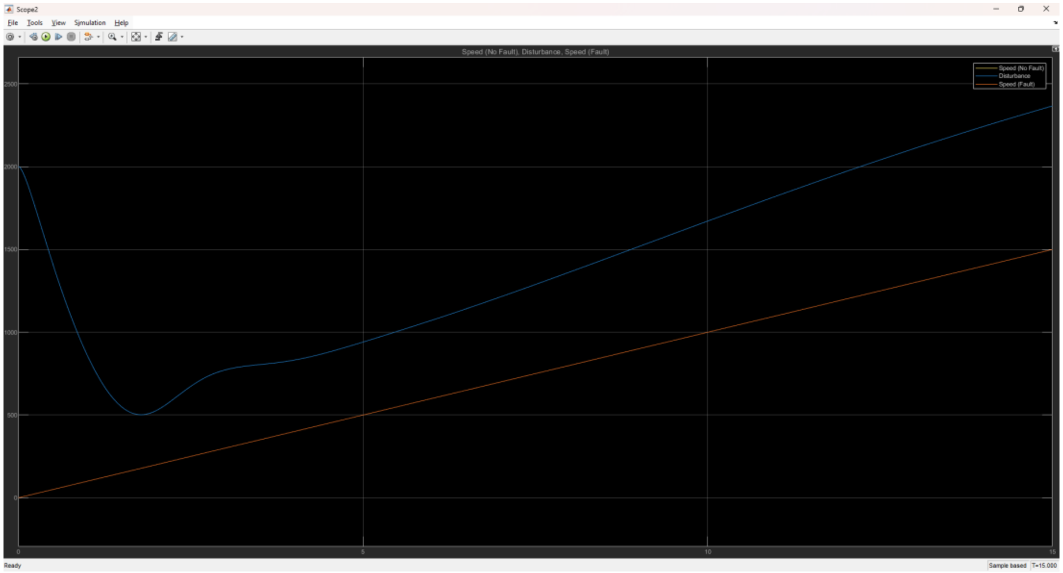


Figure 57. Throttle and Speed with no fault

The load fault disturbance is applied to the vehicle speed. The disturbance data is shown in blue signal and it increases and lowers the vehicle speed. This is the expected result as wheel faults will hinder the vehicle's speed.

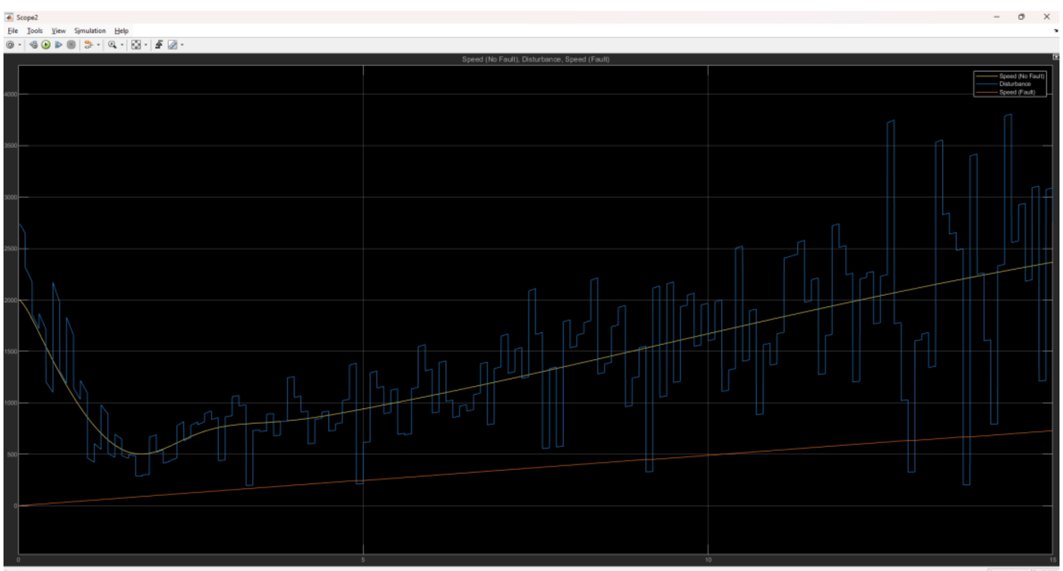


Figure 58. Throttle and Speed with Load Fault

This fault simulates when the weights mounted on the wheel falls off and now becomes unbalanced. The wheel will wobble and vibrates and hinders the vehicle to speed up. The disturbance magnitude is higher as seen with blue signal in the graph below.

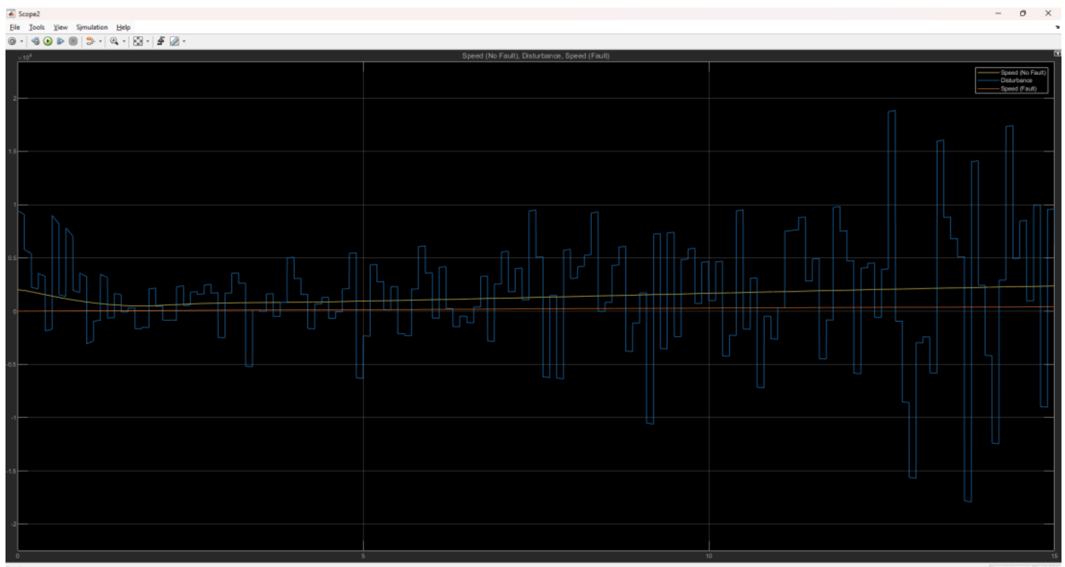


Figure 59. Throttle and Speed with Unbalanced Fault

A bearing fault is when a vehicle wheel bearing is damaged and pieces of the bearings are moving around inside the wheel while it rotates. This causes extra vibration and load to the wheel and causes issues to the vehicle when trying to speed up.

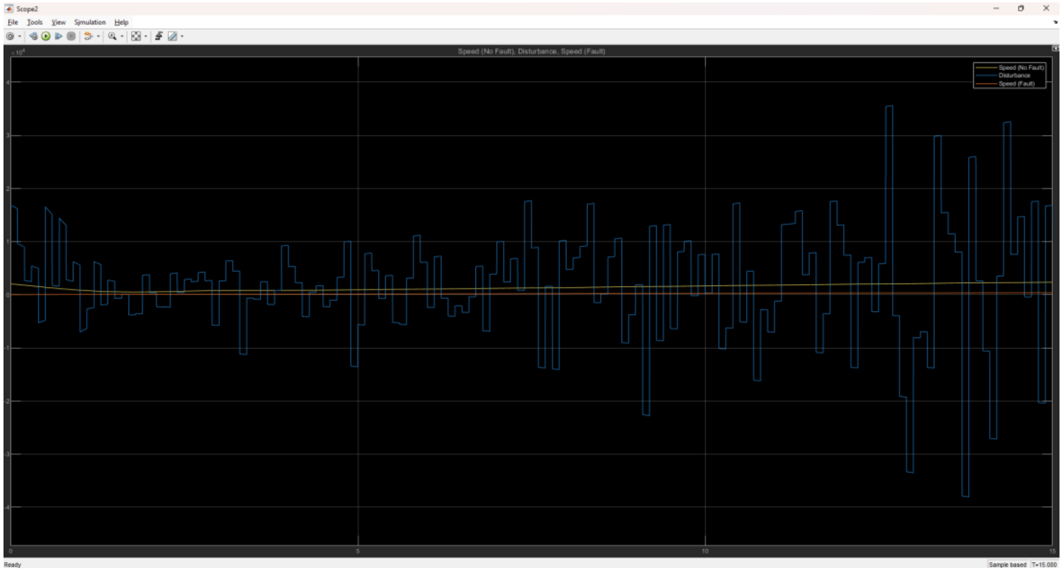


Figure 60. Throttle and Speed with Bearing Fault

Mount fault experiments with the scenario where the lugnuts that are on the wheel are damaged or not installed. Because one or more lugnuts are missing, the remaining lugnuts will bear the load as the wheel rotates. The higher the speed of rotation, there will be more load to the remaining lugnuts and eventually they will break or fall off the wheel. This is one of the highest magnitude of wheel fault and disturbance to the moving vehicle and rotating wheel.

The effects on the vehicle speed can be seen in the graph below.

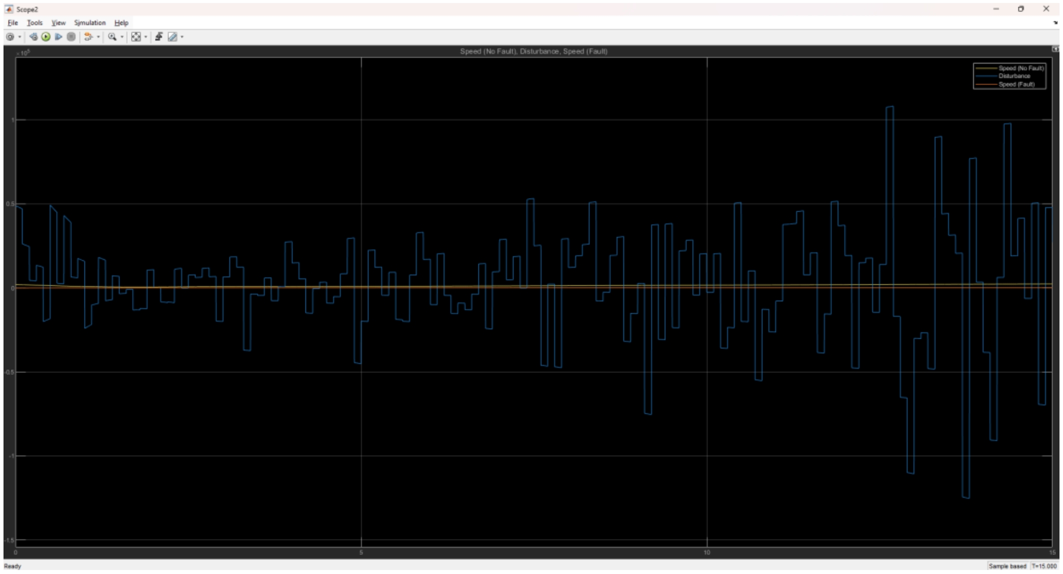


Figure 61. Throttle and Speed with Mount Fault

4.7. Detailed Analysis of Results

Simulation Results on Live Predictive Algorithm Configuration of Spectral Feed: As shown in this example the Extract Spectral Features Live Editor task can prove useful for several different applications. With the Live Editor task, you can easily match spectral peaks with known machine component frequencies. This helps you better understand your data and the mechanical components causing various features in the data.

Another application of the Extract Spectral Features Live Editor task is to generate metrics that characterize your spectral data in the frequency ranges of interest. The task produces an output table containing the peak amplitude, peak frequency, and band power of each fault frequency band, as well as the total band power of all fault bands. However, these metrics are specific to the power spectrum data input in the task.

To extend this use so that you can track metrics over time as new data sets are gathered, you can either update the power spectrum data in the task or use the automatically generated MATLAB code to produce the metrics table. Copying the MATLAB code generated is an easy way to continue computing fault band metrics for many new data sets.

A third use of the task combines the benefits of the two previously discussed uses. Since the task associates the various mechanical components with peaks in the spectral data, you can quickly determine which components cause significant changes in the spectral data and thus potential failures. For example, one common indicator of machine failure derived from spectral data is a change in the amplitude of spectral peaks. In the spectral metrics table, if you notice a significant drop in peak amplitude or band power over time you can trace the corresponding peak frequency back to the plot to see which component's fault frequency band aligns with that peak.

MATLAB command would inspect the spectrum frequencies of healthy motor data and faulty motor data and map the frequencies out onto the figure for comparison.

4.8. Final Decision of Fault Detection

The weighted average as described in the equation, the $L_A,\ L_B,\ L_C,\ \cdots$ are the resulting probability of fault from A to F. The higher the value, the more likelihood that is the true fault than other lower value faults. For example, if L_A has the highest value then this would indicate that this is most likely a fault caused by a bad battery.

4.9. Results Analysis

The simulation results are calculated using the minibatchpredict tool. This tool provides the necessary predict actions on the captured data and provides the result. The result indicates how well the trained AI tool can recognize the input data and compare the features to classify and predict its data.

Visual images of the vehicle instrument panel are recorded and processed with VisFN. The trained network can recognize and differentiate the features in faults by 65%.

The mini-batch predictions by looping over the instrument panel images resulting in a confusion matrix are shown in the figure below.

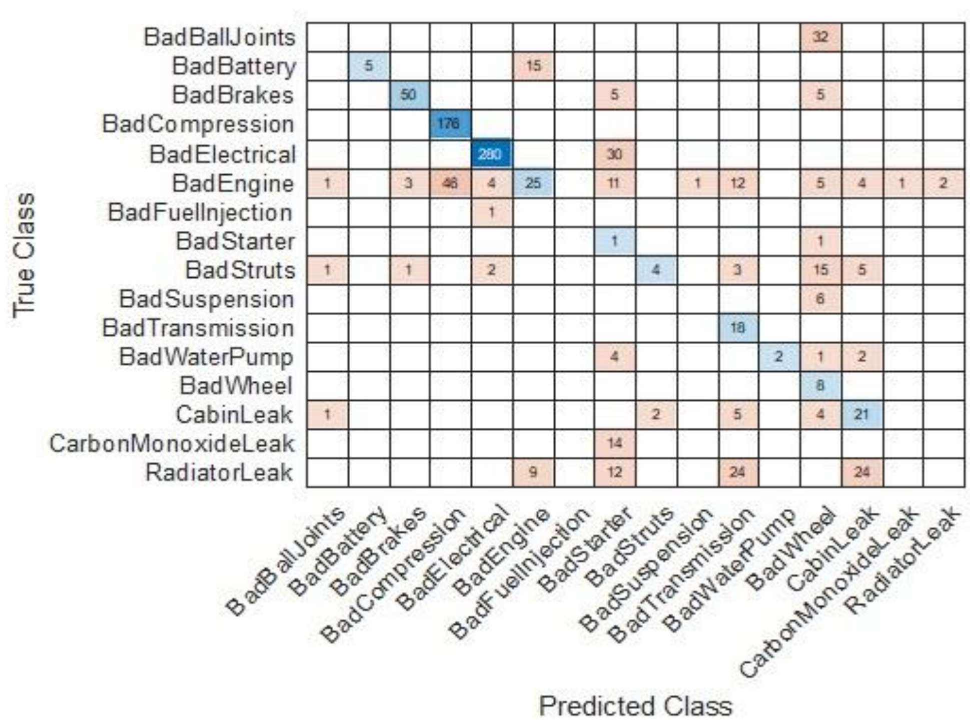


Figure 62. Validation Results of VisNET using GoogleNet

Audio sounds of the vehicle's front driver side are recorded and processed with AudFN.

The trained network can recognize and differentiate the features in faults by 68%.

The mini-batch predictions by looping over the recorded audio, resulting in a confusion matrix are shown in the figure below.

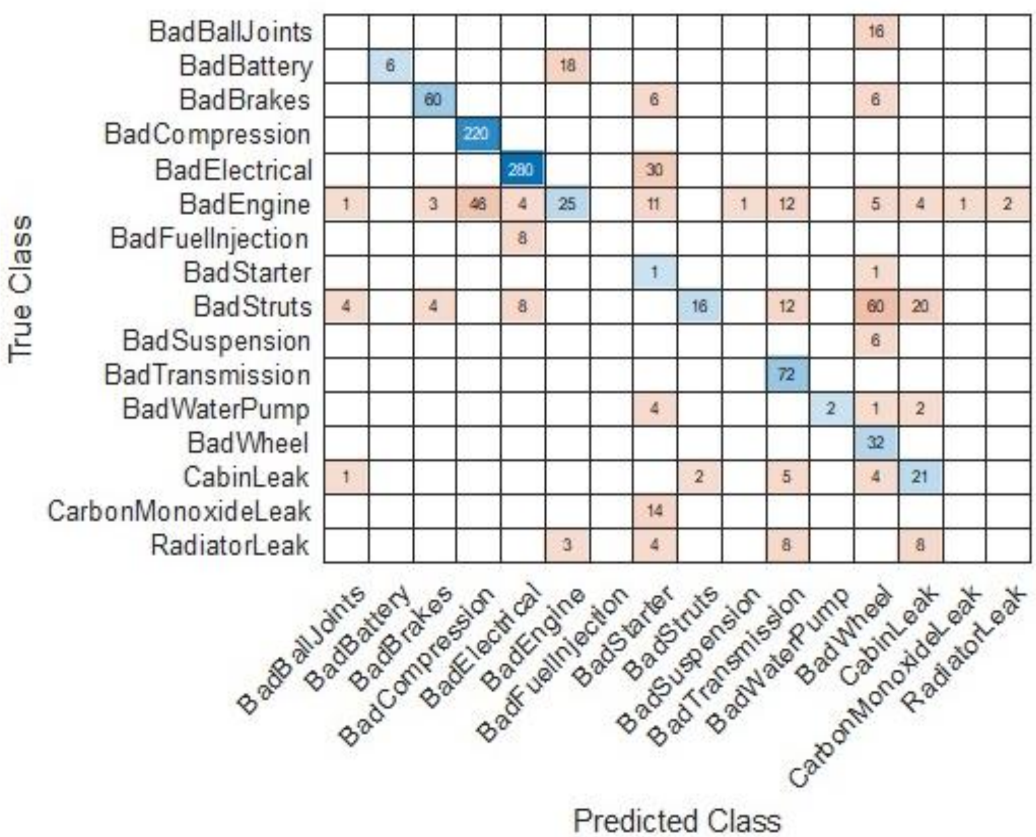


Figure 63. Validation Results of AudFN using YamNet

3-axis vibration and gyroscopic data from the strut assembly of the vehicle's front driver side are recorded and processed with VibFN.

The trained network can recognize and differentiate the features in faults by 60%. The mini-batch predictions by looping over time series axis data, resulting confusion matrix are shown in the figure below.

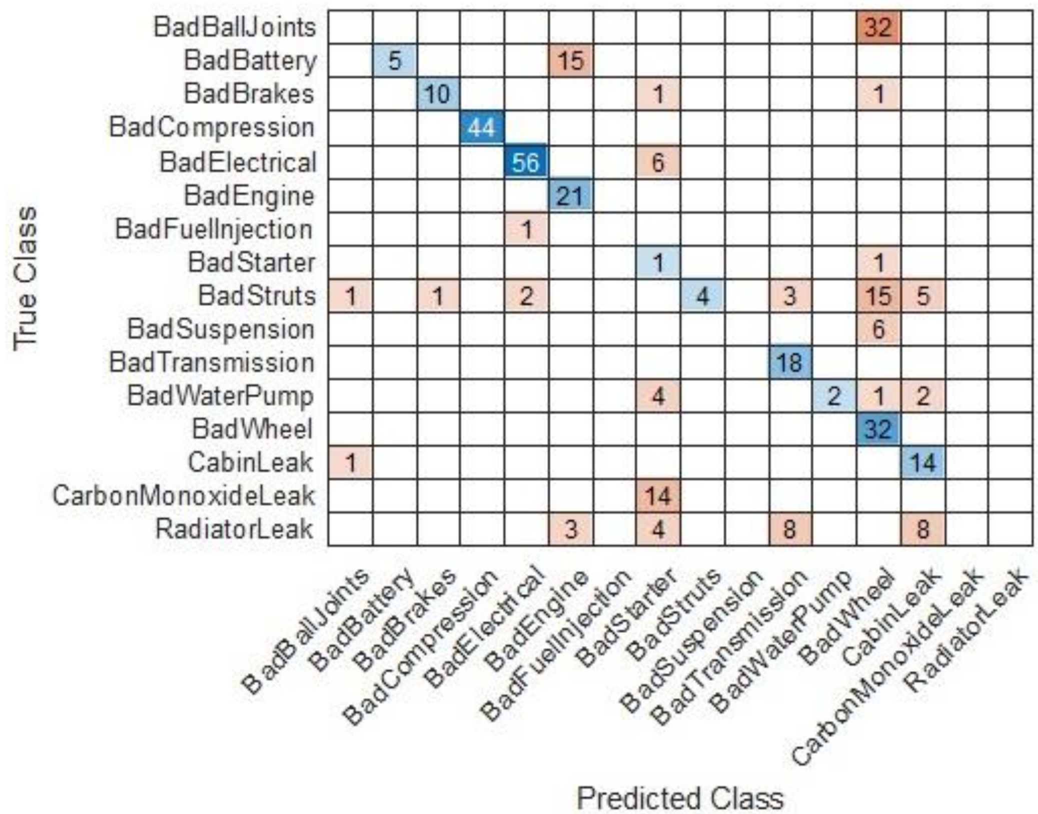


Figure 64. Validation Results of VibFN using LSTM

4.10. Wheel Fault Classification Results

The inferred vehicle wheel fault classification results are computed and summarized for faults “A”, “B”, “C”, and “D”.

The sequences of vibration values are analyzed and the sequence features are recognized by the Vibration Network with predicted scores of 66.7% for mount fault and 75% for bearing fault from the predicted class as shown below.

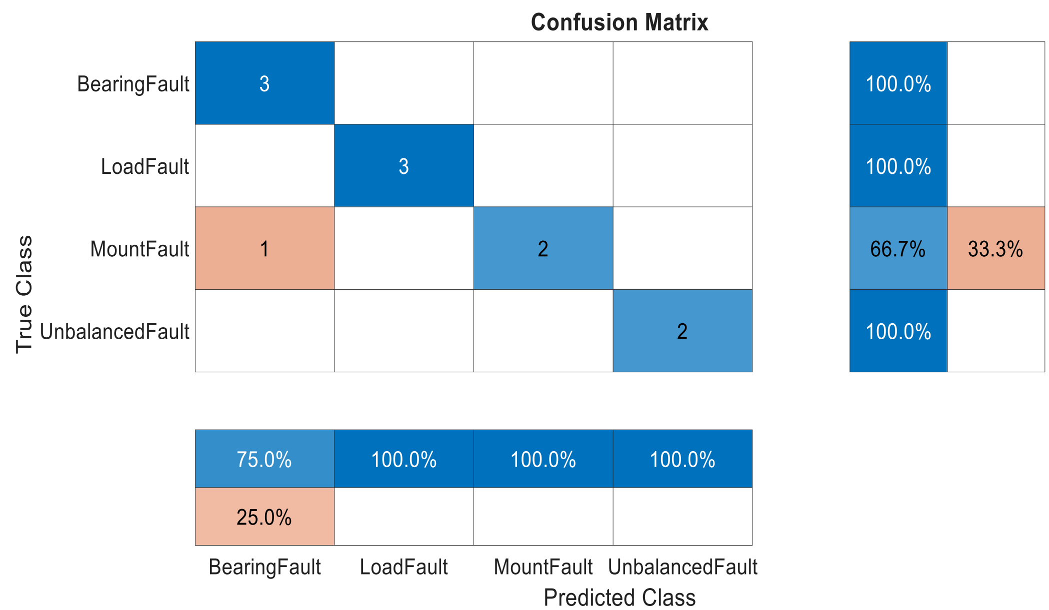


Figure 65. Wheel Fault Vibration Classification Results

The spectrum values from the audio data are analyzed and the features are recognized and classified by Audio Network with load fault values of 51.7% and 96.4% predicted class as shown below.

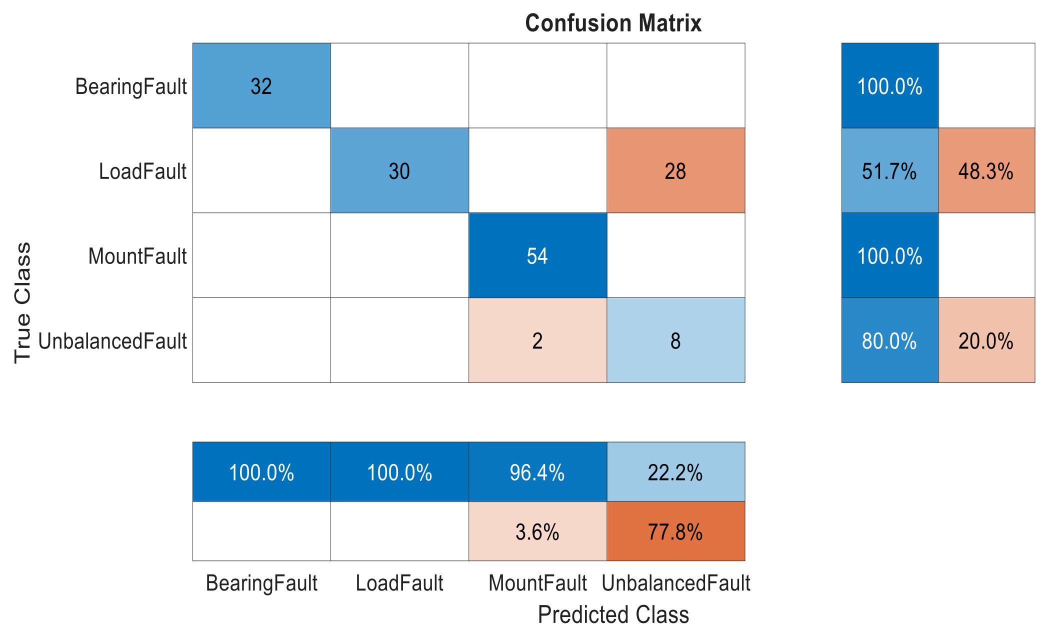


Figure 66. Wheel Fault Audio Classification Results

The visual frames of the IP are analyzed and sequenced and the classification results are 30.8% prediction as shown below.

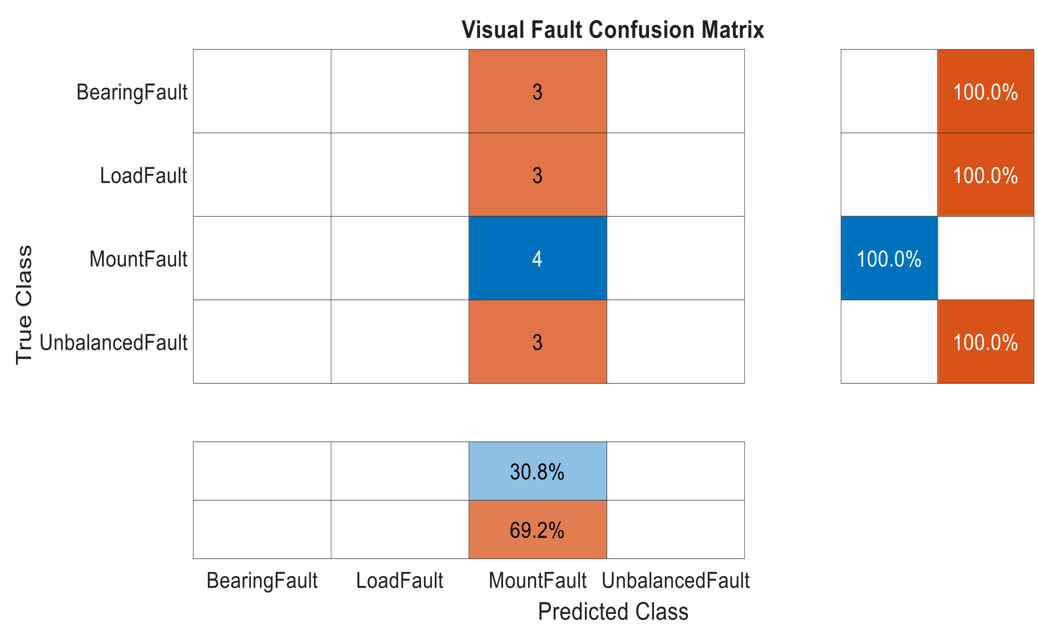


Figure 67. Wheel Fault Visual Classification Results

The FIS computed likelihood of faults with wheel mounting fault “D” is shown below.

Table 16. Wheel Fault Likelihood Results

Likelihood	L_A	L_B	L_C	L_D
Scores	0.2	0.1	0.2	0.5

The highest likelihood value of 50% correctly indicates the wheel mounting fault symptoms of L_D processed by FIS.

CHAPTER FIVE

CONCLUSION

5.1. Summary of Findings

The validation of the NIFDT system proposed by this paper shows recognition of fault classification of “A” Faulty Battery and “E” Faulty Wheel. The highest number of 89 probability of correct classification using the camera and other devices validated that this system could detect faults in its recorded features.

The challenging aspect of the NIFDT system is locating the best installation location for recording devices. No part is opened or modified, and the recording devices are installed while the vehicle is operational. Depending on the location of the device, there were distortions and undesired noise to the recorded data. This has led to outputting incorrect results and required filtering and calibrations.

Future actions would include more devices, and AI tools collecting more data on other fault types and training the AI network models to detect and recognize the faults in real-world cases. There is a lack of recorded data on machine faults and collecting and sharing this would advance the diagnostic field of NIFDT.

5.2. Key Research Outcomes and Operation Benefits

The key research outcome from this study is a new technique to detect faults and classify them by fuzzifying the prediction scores from various AI tools and using the fuzzification inference tool to fuse the results to get better predictability outcomes. There was room for improvement because the prediction results showed that prediction is not always correct. So instead of using the output results from the different AI tools, this study went one step further and recovered the prediction scores of each classification and used it to get better results by calibrating with the fuzzy tool.

This is possible because each sensor would provide different values depending on what medium it is getting the data from. For example, the visual camera records the images of the instrument panel and IMU records the vibration on the wheel and engine. Each recorded data would complement each other because when there is minimum vibration, the images of the instrument panel would have higher weights towards the output. This would improve the outcome by calibrating the weights and bias in the fuzzy tool set.

5.3. AI Integration

As different kinds of AI tools are used in this study, from image recognition tool CNN to time sequence recognition tool of RNN and LSTM, and audio recognition tool Yamnet, it requires different techniques to gather and calibrate the data for each AI tool. Each AI tool requires specific data types, length and format.

The AI integration of CNN architecture AI tool Googlenet and RNN architecture AI tool LSTM required combinations of two different structures. The first layers of the CNN would have normal visual recognition using the kernel matrix to detect edges, lines, and other image characteristics. As the image classifications are completed and saved in the matrix, now begins the sequencing. Each recorded data are now added with sequences so that the AI tool can compare the frames of images and know if it fits with any of the trained data. The connection stage of the two AI tools is how this study's Hybrid AI tool worked to recognize and classify sequenced images of the instrument panel and classify them for vehicle faults.

5.4. Comparison with Existing Methods

Compared to the existing method of using the output from the classification AI tools could provide false positives and incorrect results. This is because the AI classification tools are not perfect. The results could be incorrect due to trained data being poor, or due to insufficient data, or other reasons. To improve the classification probability of being correct, in this study have experimented using threshold and fuzzification techniques.

The threshold technique is used to minimize the false positives by assigning a designated numerical value for fault type and if the prediction scores from the classification between different faults are not more different than the threshold value of 5% then we could disregard this result. For example, if the prediction scores of the vehicle starting fault of ASD fault and Battery fault are very similar with values of ASD fault 0.31 (31%) and Battery fault 0.32(32%), this would be disregarded. The assigned threshold value is 5% and the difference between 31% and 32% is less than the threshold value. This could indicate that the trained AI tool could not differentiate the faults enough to have enough confidence in the result and could be a false positive. To reduce the number of false positives, this result is not counted.

In addition to the threshold technique, this study experimented with the fuzzification method. Similar to the threshold technique, this study would use the prediction scores instead of the single output from the classification of true or false stating whether it is a fault or not. This way we could merge the different scores from different sensors and get better completed classification results.

5.5. Research Contributions and Significance

The research contribution is data collected using real vehicles and recorded techniques and experimental results with different AI tools. The study shows the limitations of the AI tools and areas that could be improved using techniques discussed in this study to get better classification results.

Also, the technique and inference calculations implemented in this study as the surrogate model would provide data and experimental methods to study the forced disturbance effects on rotating vehicle wheels. Without a proper testing facility, vehicle, hardware, and resources, it would not be practical to intentionally cause faults to a moving vehicle due to safety concerns.

The idea is that with AI-driven sensors, this system framework can detect events. The limitations of AI tools seen in previous chapters show that it is almost impossible to get perfect classification and predictions due to the generalization requirements. Using the Fuzzy logic tool to adjust and calibrate the fuzzy values from the sensors, is my method to increase the system's classification accuracy.

5.6. Limitations of the Study and Experimental Constraints

This study lacked the resources to carry out real experiments with actually damaged machines and parts. The key to training AI tools is having a proper and abundant amount of real data. The data collected from real vehicles is very complex as it is comprised of complex moving parts and machines and to collect data with faults would require damaging the vehicle. The data would need to be collected not only after damaging the vehicle but also before it was damaged. This would provide the necessary regression data required to not only detect the faults but also be able to perceive the fault before occurring.

To overcome this lack of resource issues, this study experimented with techniques to improve the results with threshold and fuzzification to calibrate the data and minimize the uncertainty.

5.7. Scalability and Domains

This study experimented with vehicles and motors to test the capability of recognizing the defects and faults that could be captured with visual, audio, and vibration data. But this could be scaled for more complex machines and other domains. Machines, equipment, and hardware tools used in the medical field, industrial plants and manufacturing could benefit from the idea of non-invasive fault detection with AI, NIFDT. The general concept is the same whether it is experimented on a real vehicle or a robotic arm in surgery or a manufacturing site. The NIFDT supplements the limitation of the if parts are no longer working inside the machine, the LED blinks. For example, the diagnostic system inside the vehicle will only inform the driver after the vehicle is damaged or part is damaged alerted by an internal sensor. But if we look at the combined signs and operation from overview, using NIFDT could perceive the subtle signs of ongoing issues and prevent further faults whether it is one machine or fleets of machines.

The fault detection is one of the many application that can be realized with this framework of detecting events. This could be scaled to any framework and system that could benefit from monitoring fuzzy values and anamoly detection in large scale.

5.8. Final Thoughts and Future Outlook

The essence of this study is fault detection of subtle telltale signs that are emitted outside the system. This could be sound that uses air as a medium and connected hardware objects' vibration that could be felt and measured. Because the data are too complex to detect with structured algorithms and signal processing equations, a generalized AI tool is used to get the batch-processed algorithm as an output. This was not possible in the past due to resource constraints with processing power but as technology has advanced and calculation and processing efficiency have increased significantly, AI tools have the necessary toolset required to improve and offer more practical functionalities to detecting faults and predicting the regression with precision and high level of true results.

Machines would be able to be repaired before damaged and fleets of vehicles will be monitored and maintained remotely by large numbers. The maintenance of machines will not focus on time intervals but on maintenance efficiency and proactive extension of usage. The future outlook for maintenance would be autonomous and would not require supervision and intervention by humans.

REFERENCES

1. Josephine Wolff, "How Is Technology Changing the World, and How Should the World Change Technology?", Section: Technology and Global Change, Global Perspectives (2021) 2 (1): 27353., https://doi.org/10.1525/gp.2021.27353

2. R. Paul, et al., "Conceptualisation of a novel technique to incorporate artificial intelligence in preventive and predictive maintenance in tandem," Materials Today: Proceedings, vol. 66, Part 9, 2022, pp. 3814-3821, ISSN 2214-7853, [Online]. Available: https://doi.org/10.1016/j.matpr.2022.06.250

3. J. Dalzochio, et al., "Machine learning and reasoning for predictive maintenance in Industry 4.0: Current status and challenges," Computers in Industry, vol. 123, 2020, 103298, ISSN 0166-3615, [Online]. Available: https://doi.org/10.1016/j.compind.2020.103298

4. A. Reñonesa, et al., "F.A.I.R. open dataset of brushed DC motor faults for testing of AI al-gorithms," ADCAIJ: Advances in Distributed Computing and Artificial Intelligence, vol. 9 N. 4 (2020), 83-94 eISSN: 2255-2863, [Online]. Available: https://doi.org/10.14201/ADCAIJ2020948394

5. Mubarak Alotaibi, et al., "Non-Invasive Inspections: A Review on Methods and Tools," Sensors (Basel). 2021 Dec; 21(24): 8474, [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8705398/

6. Thorsten Wuest, et al., "Machine learning in manufacturing: advantages,challenges, and ap-plications.", ISSN: (Print) 2169-3277, [Online]. Available: https://www.tandfonline.com/loi/tpmr20

7. Khanh T.P. Nguyen, et al., "A new dynamic predictive maintenance framework using deep learning for failure prognostics.", [Online]. Available: https://www.elsevier.com/locate/ress

8. Amrit Raj Paul, et al., "Conceptualisation of a novel technique to incorporate artificial intelli-gence in preventive and predictive maintenance in tandem.", Materials Today: Proceedings, journal, [Online]. Available: https://www.elsevier.com/locate/matpr

9. Majdi, H. S., Shijer, et al., "Analysis of fault diagnosis of DC motors by power consumption pattern recognition.", Eastern-European Journal of Enterprise Technologies, 5 (5 (113)), 14–20. doi: https://doi.org/10.15587/1729-4061.2021.240262

10. Abderrahim Allal, et al., “Diagnosis of induction motor faults using the motor current nor-malized residual harmonic analysis method.”, International Journal of Electrical Power and Energy Systems, journal, [Online]. Available: https:// www.elsevier.com/locate/ijepes

11. S.Krishnaveni, et al., “Analysis and control of the motor vibration using arduino and machine learning model”, materials today proceedings, Volume 45, Part 2, 2021, Pages 2551-2555, https://doi.org/10.1016/j.matpr.2020.11.261

12. Shorouk OssamaI, et al., “Implementation of fuzzy modeling system for faults detection and diagnosis in three phase induction motor drive system”, Journal of Electrical Systems and Information Technology, Volume 2, Issue 1, May 2015, Pages 27-46, https://doi.org/10.1016/j.jesit.2015.03.004

13. Yanqing Duan et al., “Artificial intelligence for decision making in the era of Big Data evolution, challenges and research agenda”, International Journal of Information Management, Volume 48, 2019, Pages 63-71, ISSN 0268-4012, https://doi.org/10.1016/j.ijinfomgt.2019.01.021.

14. Andrew K.S., et al., “A review on machinery diagnostics and prognostics implementing condition-based maintenance”, Mechanical Systems and Signal Processing Volume 20, Is-sue 7, October 2006, Pages 1483-1510, https://doi.org/10.1016/j.ymssp.2005.09.012

15. Nazanin Vafaei, et al., “Fuzzy early warning systems for condition based maintenance”, Computers & Industrial Engineering Volume 128, February 2019, Pages 736-746, https://doi.org/10.1016/j.cie.2018.12.056

16. E. Smith and F. Jones, "Implementing AI-Based Surrogate Models for Vehicle Dynamics," in Proc. IEEE Int. Conf. on Intelligent Transportation Systems (ITSC), Anchorage, AK, Sep. 2023, pp. 1103-1108.

17. G. R. Davis and H. T. Nguyen, "Real-Time Optimization of AC Motors Using AI Surrogate Models," in Proc. IEEE Int. Conf. on Control Applications (CCA), Paris, France, Sep. 2024, pp. 250-255.

18. Calvo, J. & Diaz, V. & San Román, J.. (2005). Establishing inspection criteria to verify the dynamic behaviour of the vehicle suspension system by a platform vibrating test bench. International Journal of Vehicle Design - INT J VEH DES. 38. 10.1504/IJVD.2005.007623.

19. I. C. Rosca and M. Filip, "Damping ratio estimation of Three-Phase Induction Motor," 2014 International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), Bran, Romania, 2014, pp. 405-408, doi: 10.1109/OPTIM.2014.6851021.

20. Marco Muenchhof, et al., “Fault Tolerant Actuators and Drives – Structures, Fault Detection Principles and Applications.”, Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes Barcelona, Spain, June 30 - July 3, 2009

21. Purushottam Gangsar, et al., “Signal based condition monitoring techniques for fault detec-tion and diagnosis of induction motors: A state-of-the-art review.”, Mechanical Systems and Signal Processing journal, [Online]. Available: https://www.elsevier.com/locate/ymssp

22. Marcia Baptista, et al., “Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling”, Computers & Industrial Engineering Volume 115, Janu-ary 2018, Pages 41-53, https://doi.org/10.1016/j.cie.2017.10.033

23. prem shankar kumar, et al., “Bearing degradation assessment and remaining useful life esti-mation based on Kullback-Leibler divergence and Gaussian processes regression”, Meas-urement, Volume 174, April 2021, 108948, https://doi.org/10.1016/j.measurement.2020.108948

24. A.Sivakumar, et al., “Reduction of source current harmonics in ANN controlled induction motor”, Alexandria Engineering Journal Volume 57, Issue 3, September 2018, Pages 1489-1499, https://doi.org/10.1016/j.aej.2017.03.048

25. Bennett, E., & Schiller, D. (2010). On-Board Diagnostics II: A Primer. *Automotive Engineering*, 118(7), 52-56.

26. G. R. Davis and H. T. Nguyen, "Real-Time Optimization of AC Motors Using AI Surrogate Models," in Proc. IEEE Int. Conf. on Control Applications (CCA), Paris, France, Sep. 2024, pp. 250-255.

27. R. M. Brach, Structural Vibration: Analysis and Damping, 2nd ed. New York, NY, USA: Wiley, 2006.

28. M. J. Crocker, Handbook of Acoustics, 2nd ed. Hoboken, NJ, USA: Wiley, 2007.

29. S. K. Mitra, Digital Signal Processing: A Computer-Based Approach, 4th ed. New York, NY, USA: McGraw-Hill, 2011.

30. J. L. Crecraft and T. D. Dutton, Mathematical Methods for Physicists, 7th ed. Amsterdam, The Netherlands: Elsevier, 2014.

31. T. D. Gillespie, Fundamentals of Vehicle Dynamics. Warrendale, PA, USA: SAE International, 1992.

32. Jürgen Schmidhuber, “Deep learning in neural networks: An overview”, Neural Networks Volume 61, January 2015, Pages 85-117, https://doi.org/10.1016/j.neunet.2014.09.003

33. Hadis Hesabi a,c, et al., “A deep learning predictive model for selective maintenance optimi-zation.”, Reliability Engineering and System Safety, journal, [Online]. Available: https://www.elsevier.com/locate/ress