Structural Health Monitoring

Edge Impulse Deployed to Edge Hardware for Real-World Impact

Objectives of SHM Research Project

- Proof of concept through experimental trials and data analysis
- Identification of critical points in structures
- Development of a sensor network using a centralized processor, data acquisition module, and wireless communication capabilities
- Development of machine learning algorithms to detect structural changes
- Prove scalability and reliability of low power sensor network, and feasibility of using real time processing on newer Arduino architectures.

Machine Learning

Real-Time Signal Recognition

- AI: Simulates human intelligence processes
  - Pattern recognition and language processing
- ML: Machine learning is a subset of AI specializing in making predictions and learning from the data.
- Nearest Neighbor Algorithm: Plots data vectors in a variable-dimension space, and finds its nearest neighbor.
- Shallow ANN (Artificial Neural Network)
  - A neural network with only one hidden layer of neurons between the input and the output.

Purpose of ML in SHM

- Detecting **patterns and outliers** which can indicate structural weakness or strain
- Recognizing **damage** from natural disasters or the need for maintenance based on sensor data
- **Improve** based on a weighted average of data collected over time (to avoid overfit)
- **Reporting** anomalies and identifying important status changes in the structure

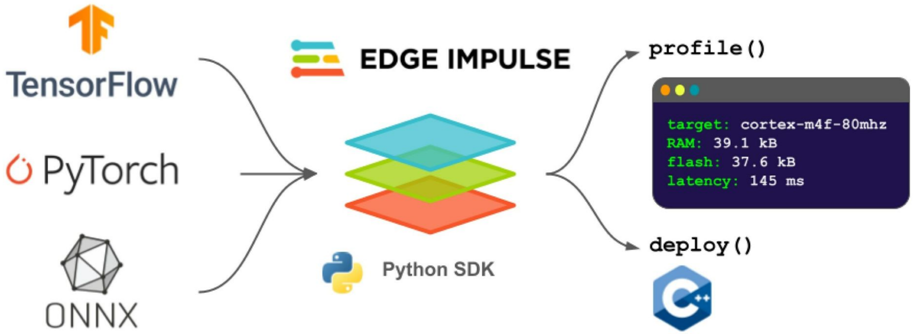
Machine Learning

Potential Lightweight Algorithms - A Solution to Limited Resources

- Thresholding - classifying events based on the amplitude sensors by setting a thresholds above a certain levels.
- Moving Averages - smooth sensor data and reduce noise
- Feature Extraction - mean, standard deviation, peak amplitude…used as input features
- Decision Trees - Train model to classify different patterns or sensor events
- Support Vector Machines - binary classification tasks
- REAL-TIME PROCESSING - CODE OPTIMIZATION
  - Low latency
  - Model complexity and hardware limitations

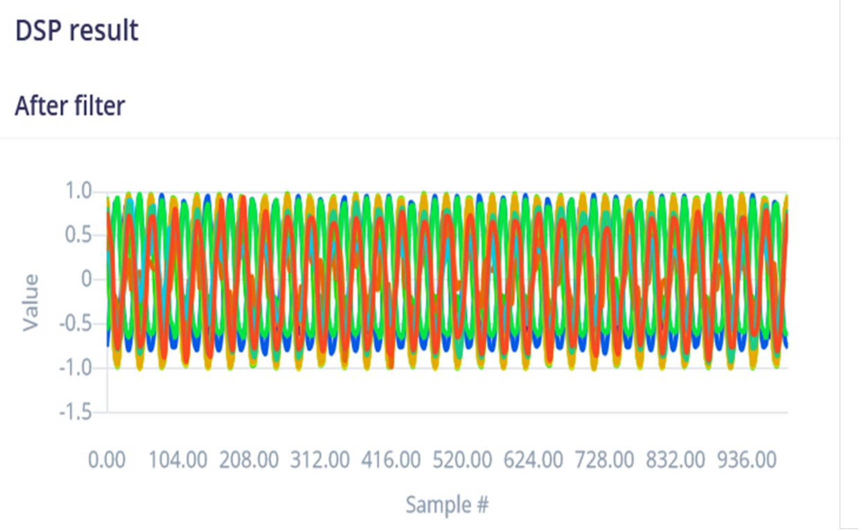
ML Service - Edge Impulse

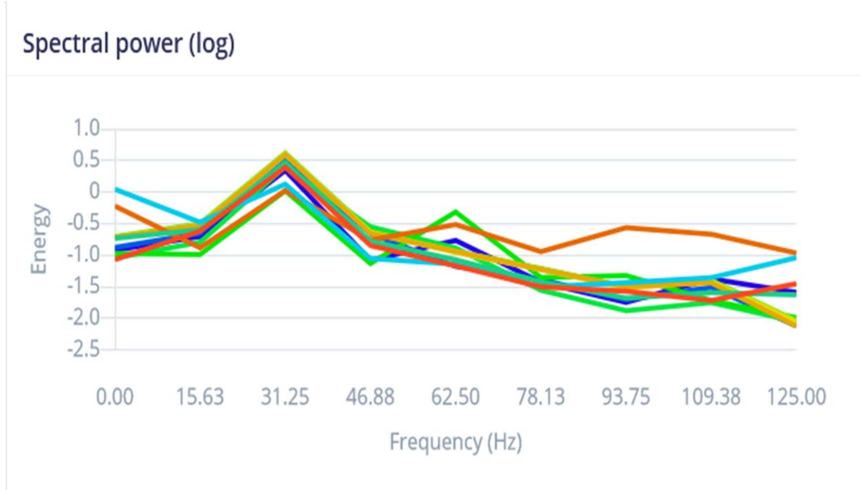
- Cloud-Based Machine Learning
- Convenient Training Method
- Drop-In Code for Arduino Giga



Our Edge Impulse Model

- Contains two hidden layers with 40 neurons total
- Supports spectral analysis and spectrogram preprocessing
- 81,000 feature variables
- ~ 60 ms latency
- 200 kB RAM
- 100 kB ROM
- Compatible with Arduino M7 processor





The project involved post processing data to extract features that differentiate the levels of damage at critical points of a structure. We used ML training software Edge Impulse that specializes in signal processing applications and deployment on programmable logic controllers like the Arduino. We processed the data through different filters, such as the spectral analysis shown. **By including parameters for intensity, frequency, and location on the structure,** our model was able to successfully create feature groupings in multi-dimensional variable space.

Alternative Signal Processing Techniques

- Statistical time series models
- Probability density function
- Cohen’s Class
- Cepstrum Analysis
- Hilbert-Huang Analysis
- Wigner Ville Distribution

Alternative Sensors

Displacement transducers

Velocity transducers

Laser Doppler Vibrometers

Acoustic Sensors

Humidity Sensors

Anemometers

Pressure Sensors

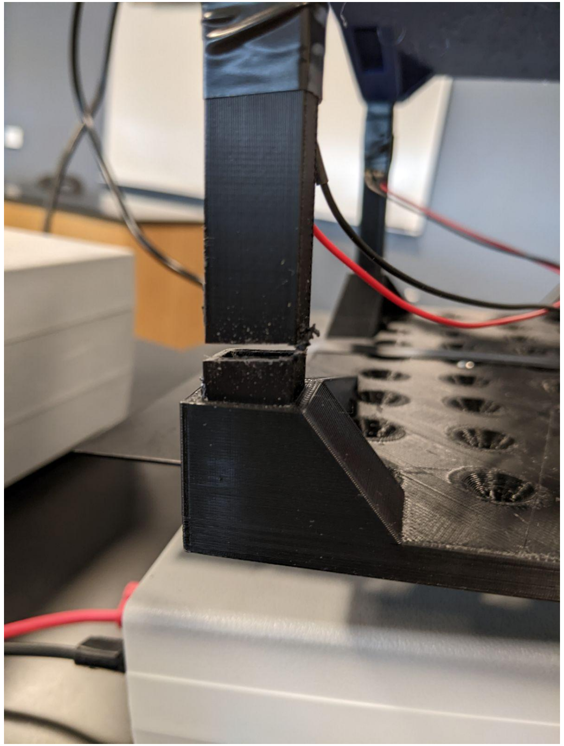
1. Project Description

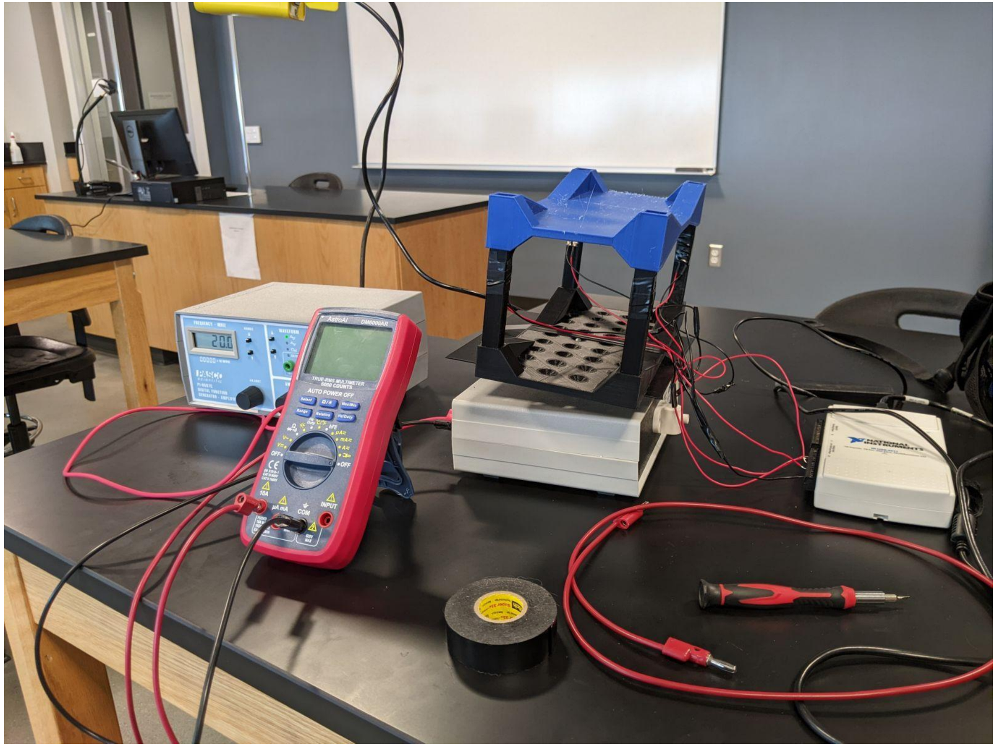
- An automated system which utilizes sensors and analytical techniques to monitor structure health.

2. Problem Statement

- How can we better assess the integrity of structures with data driven solutions
- Current SHM solutions: Costly and Complex to manage
- Our goal is to develop an integrated SHM solution with the goal of providing real-time information about the health of a structure

Testing the Strength of the Signals on a Damaged Structure (Bridge Mk. I)





Physical Setup of Container

Fig 0: Full Set Up

Physical Setup of Container

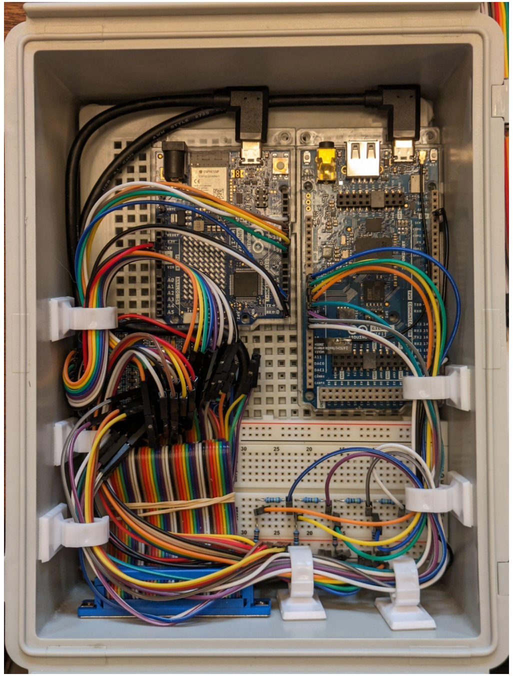


Fig 1: Processing and reading a load sensor

Structure

Fig 2: Structure base plate printing (PLA)

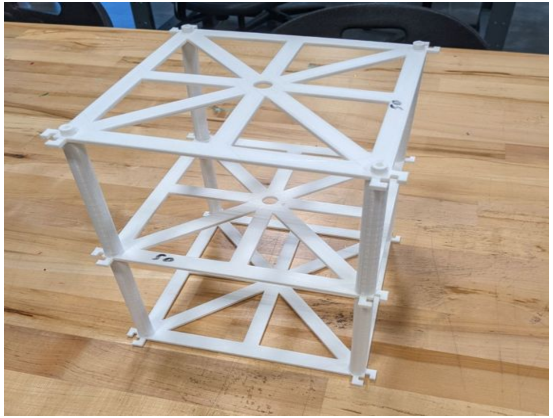
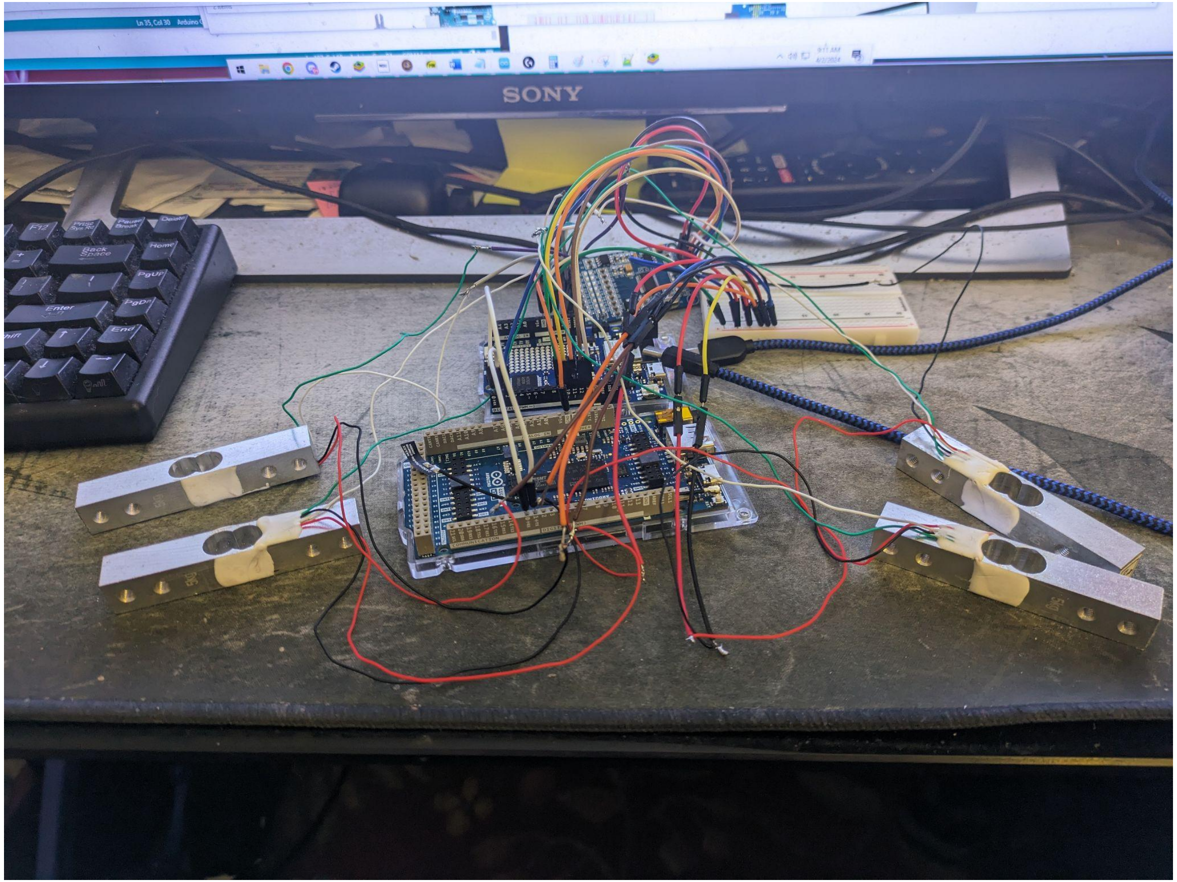


Fig 3: Structure mounted using the baseplates and interchangeable columns

Fig 4: The shaker

Fig 5: Four-tiered structure constructed.

Prototype with Digital Load Sensors



Sensors

Fig 6: Piezoelectric Sensor

Sensors used:

- 8 x Flexible piezoelectric sensors
- 1 x Vibration sensor
- 4 x 5-kg Digital Load Cell Sensors
- 1 x ADS – 1256

Microcontrollers

- Arduino Uno R4 Wifi
- Arduino Giga R1 Wi-Fi

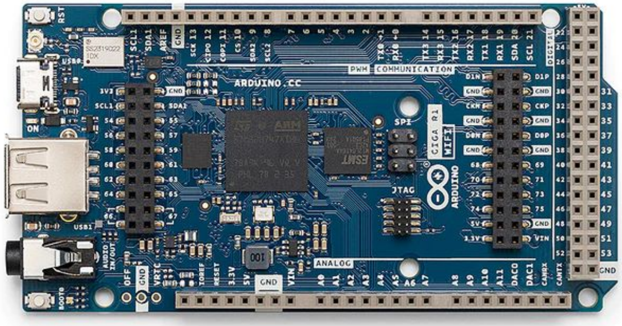


Fig7: Arduino Giga R1

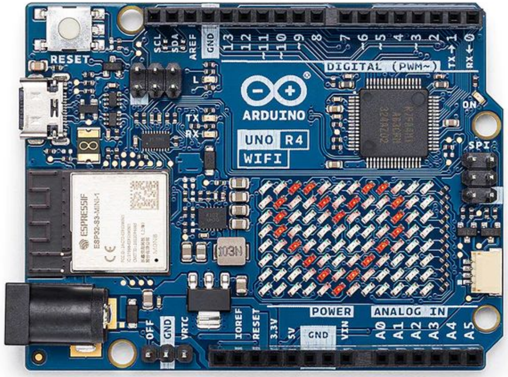
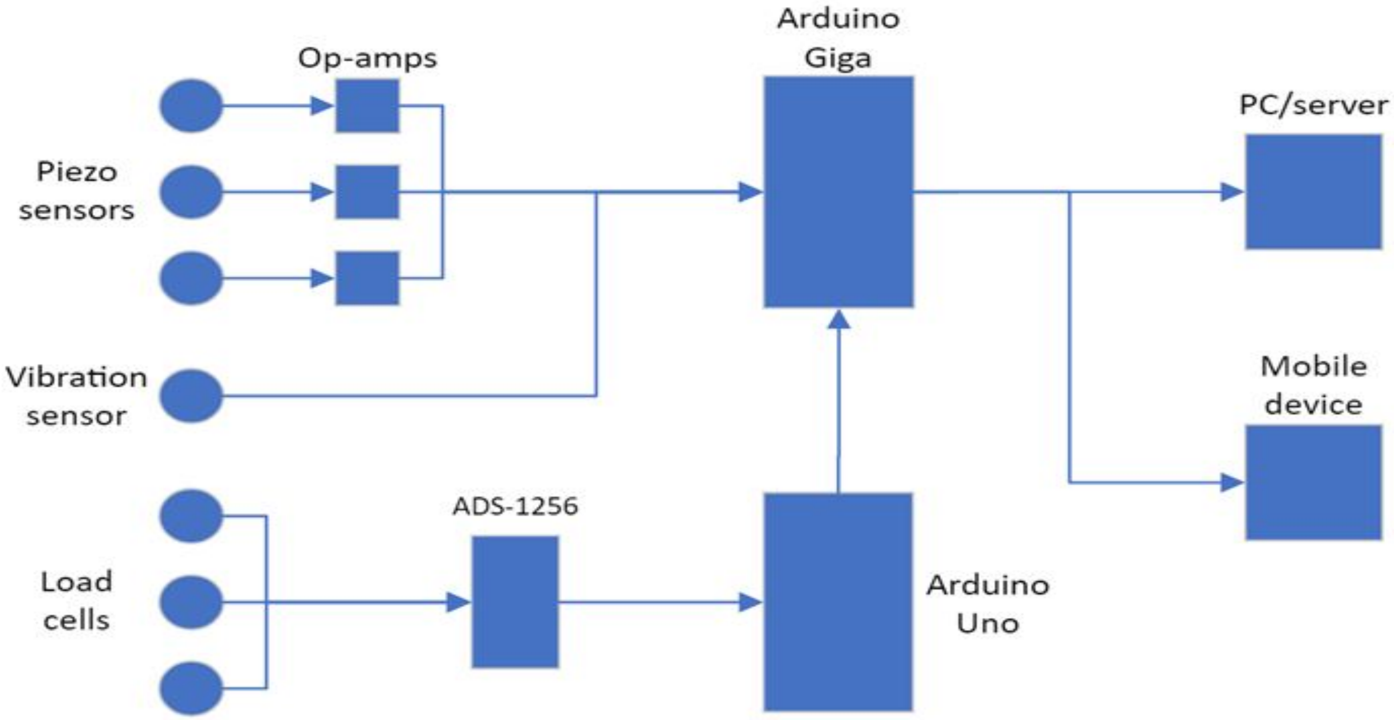


Fig8: Arduino Uno R4

Final Hardware Framework



Communication Libraries

- Between CPUs
  - RPC.begin()
  - RPC.print()
  - RPC.read()
- Between Arduinos
  - Serial.write()
  - Serial.read()

Communication between Arduinos

Communication between processors

Final Software Framework

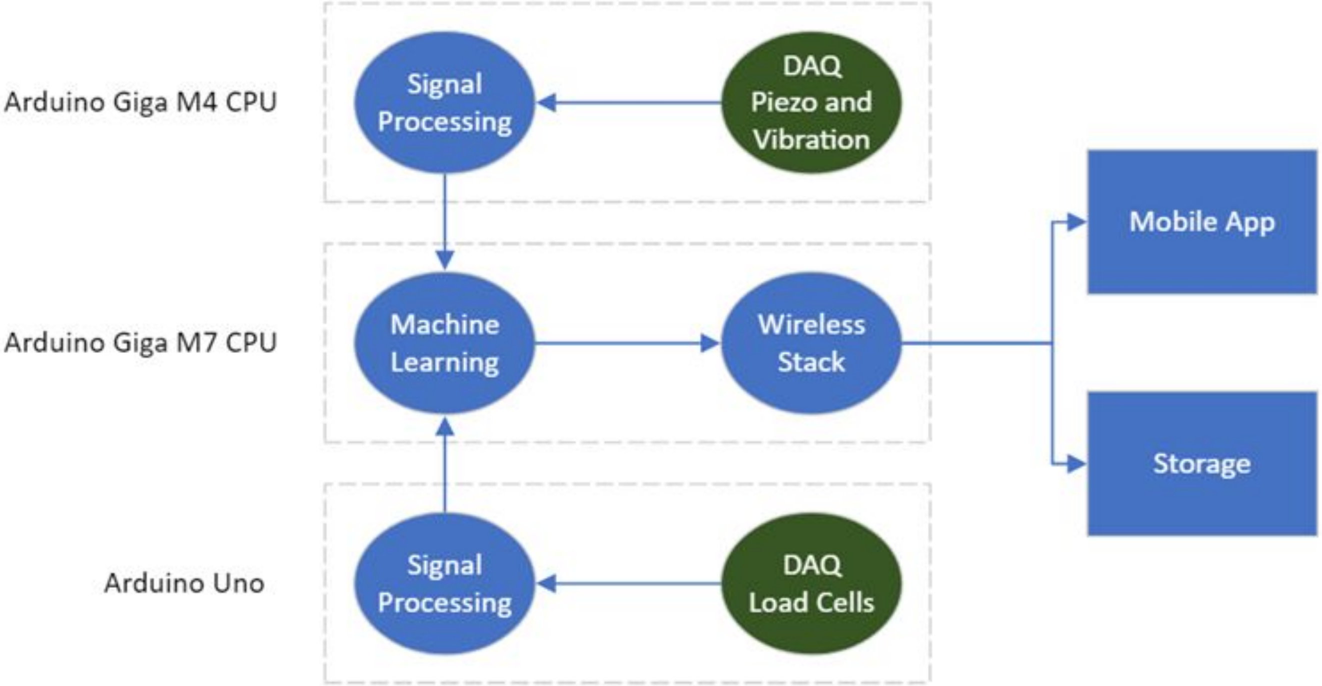
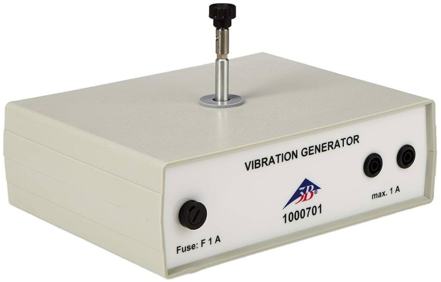
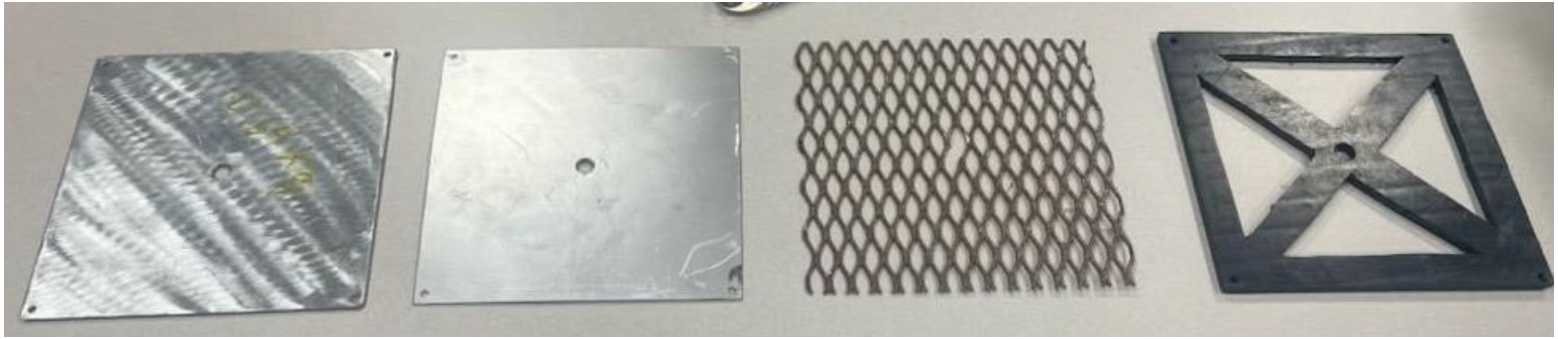


Fig10: Data Acquisition pipeline

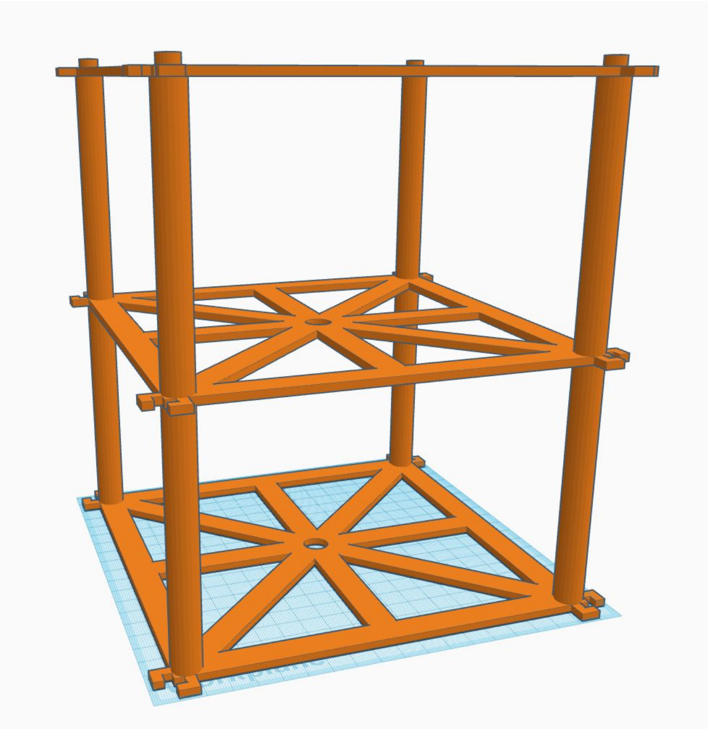
Shaker and Auxiliary Shaker Hardware





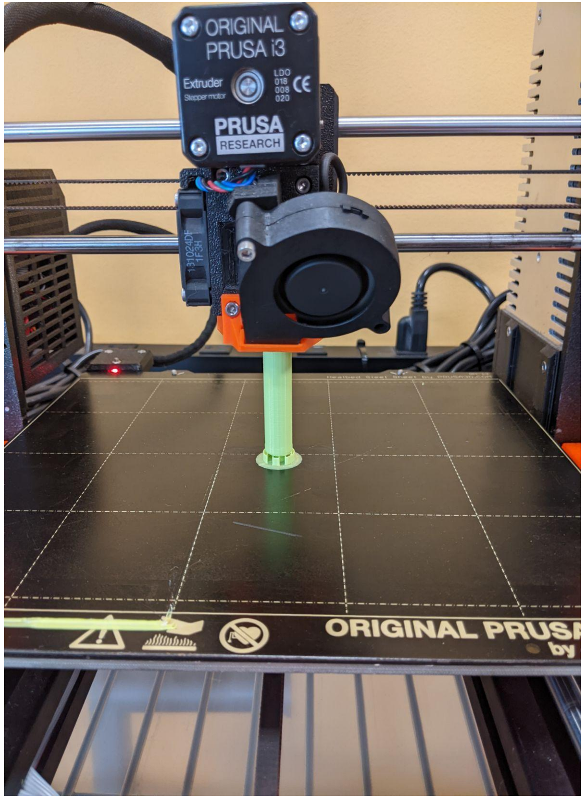
Three tier structure

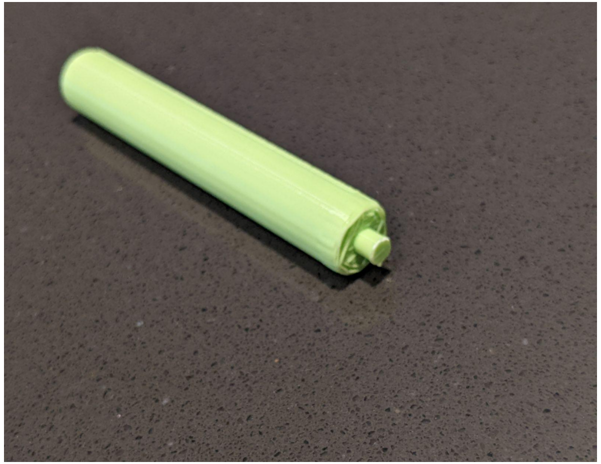
- Modular structure allows repeated damage attempts
- Hooks allow support for cross beams facilitated by rubber bands
- Capable of scaling taller or shorter as project design requires
- Was printed with a combination of PLA and TPU filament
- Skeletonization reduces mass and increases vibration amplitude



18 cm x 18 cm x 24 cm

3D Printing the Columns





DAQ devices

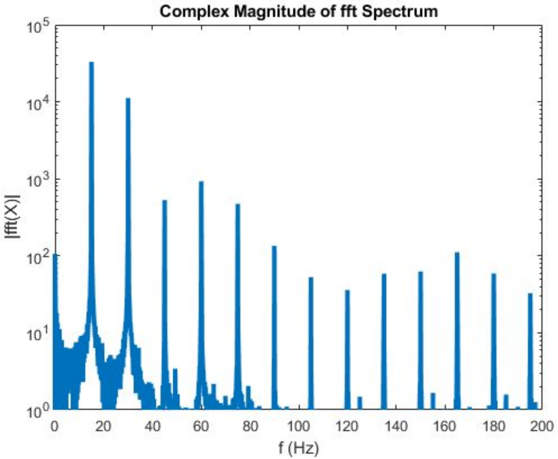
NI USB-6211

NI USB-6003

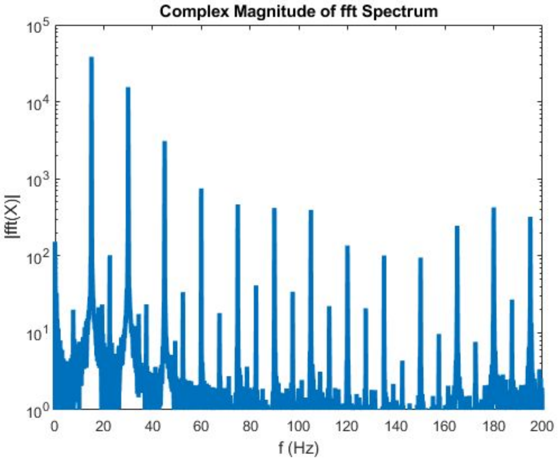
Signal Processing

- Fast Fourier Transform (FFT)
- Short-time Fourier Transform (STFT)
- Power Spectral Density (PSD)

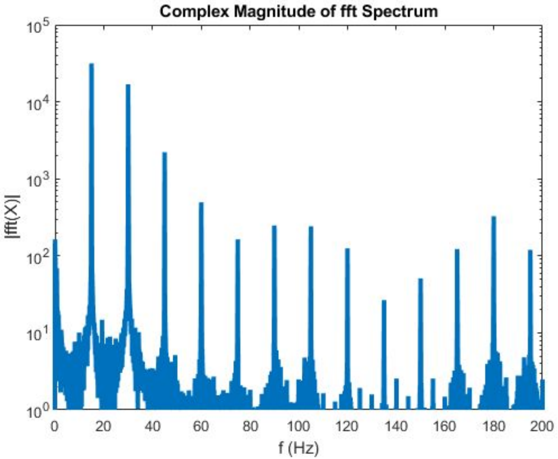
FFT Analysis



Undamaged
FFT

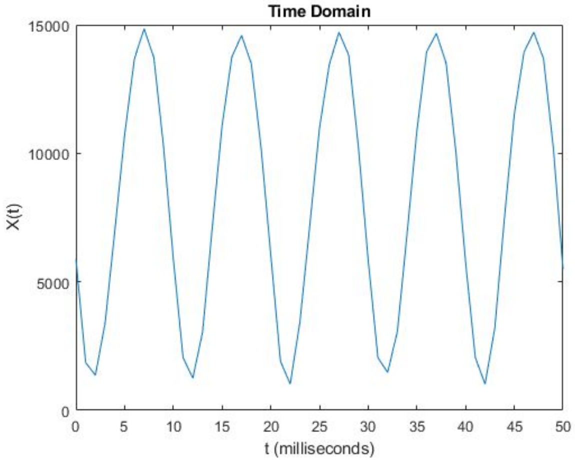


Damaged Crossbeam
FFT

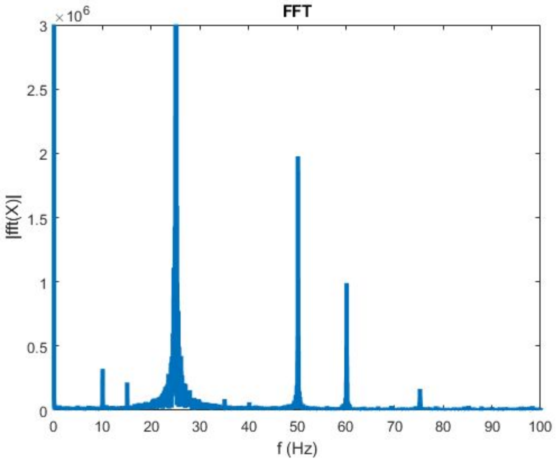


Damaged Column
FFT

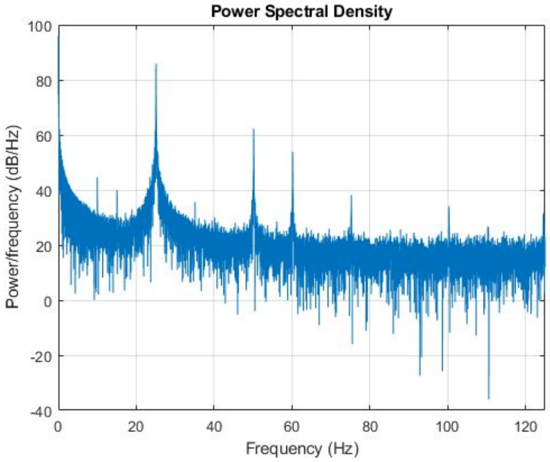
Domain Representations



Time Domain

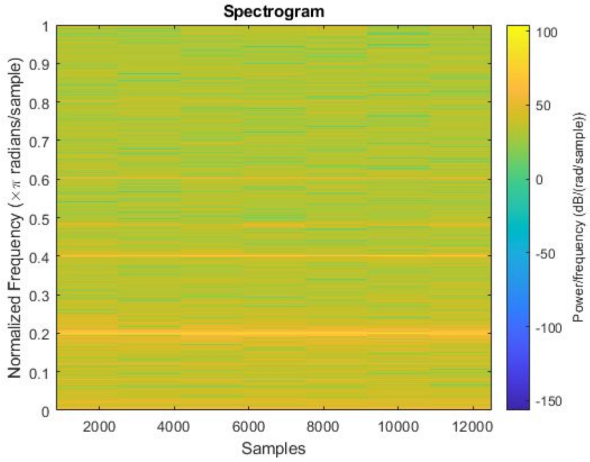


FFT

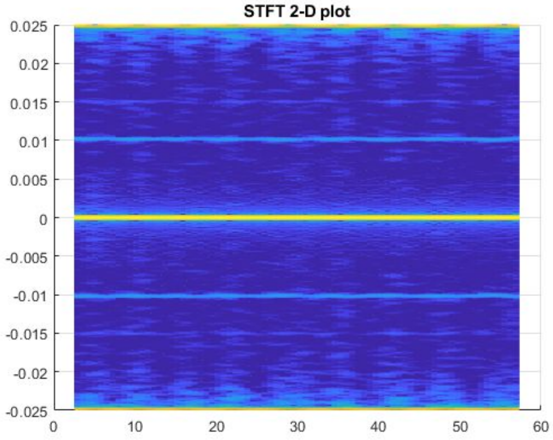


Power Spectral Density

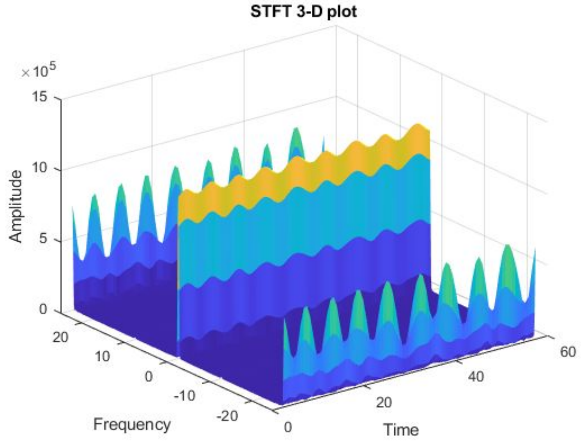
Visualization and STFT



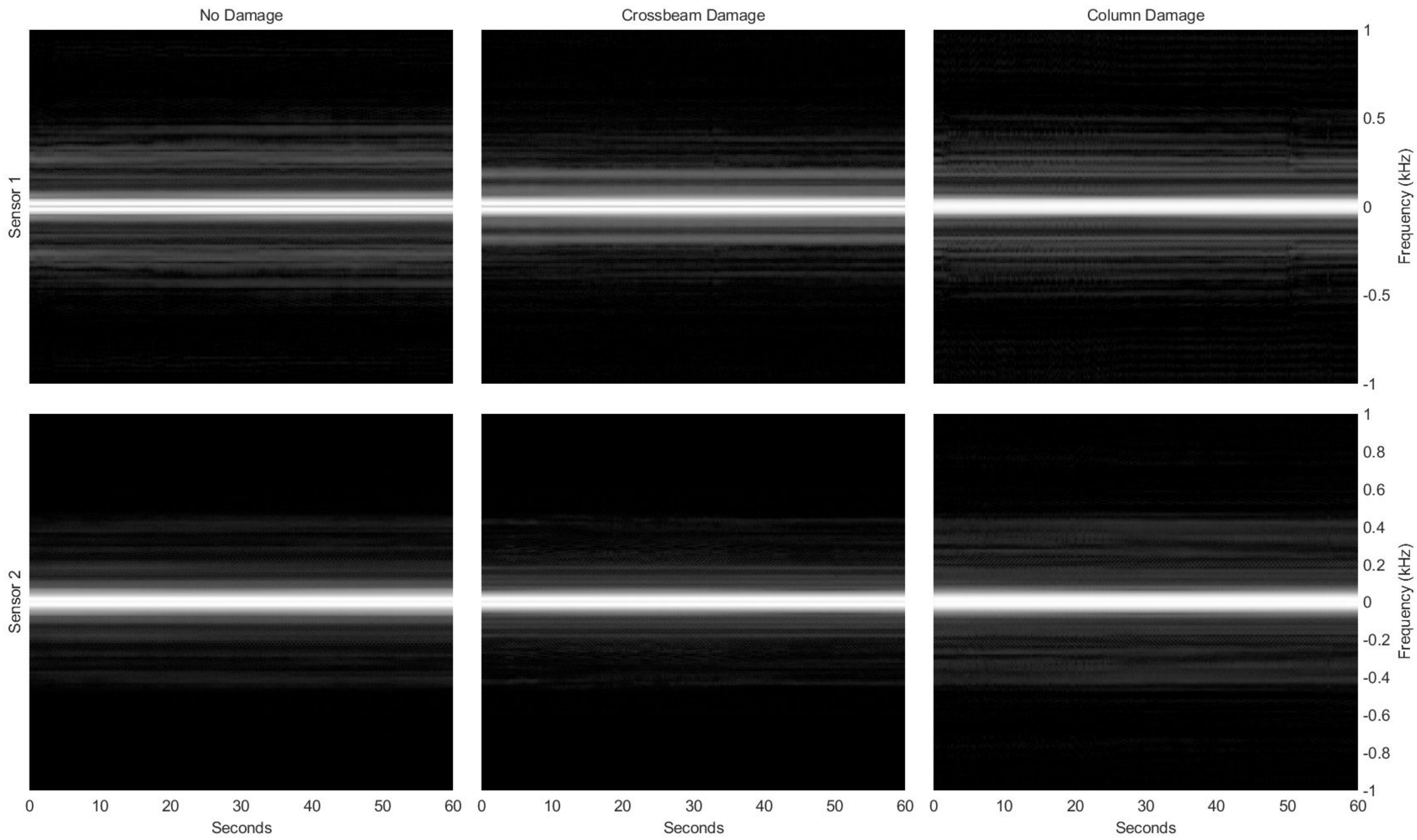
Spectrogram

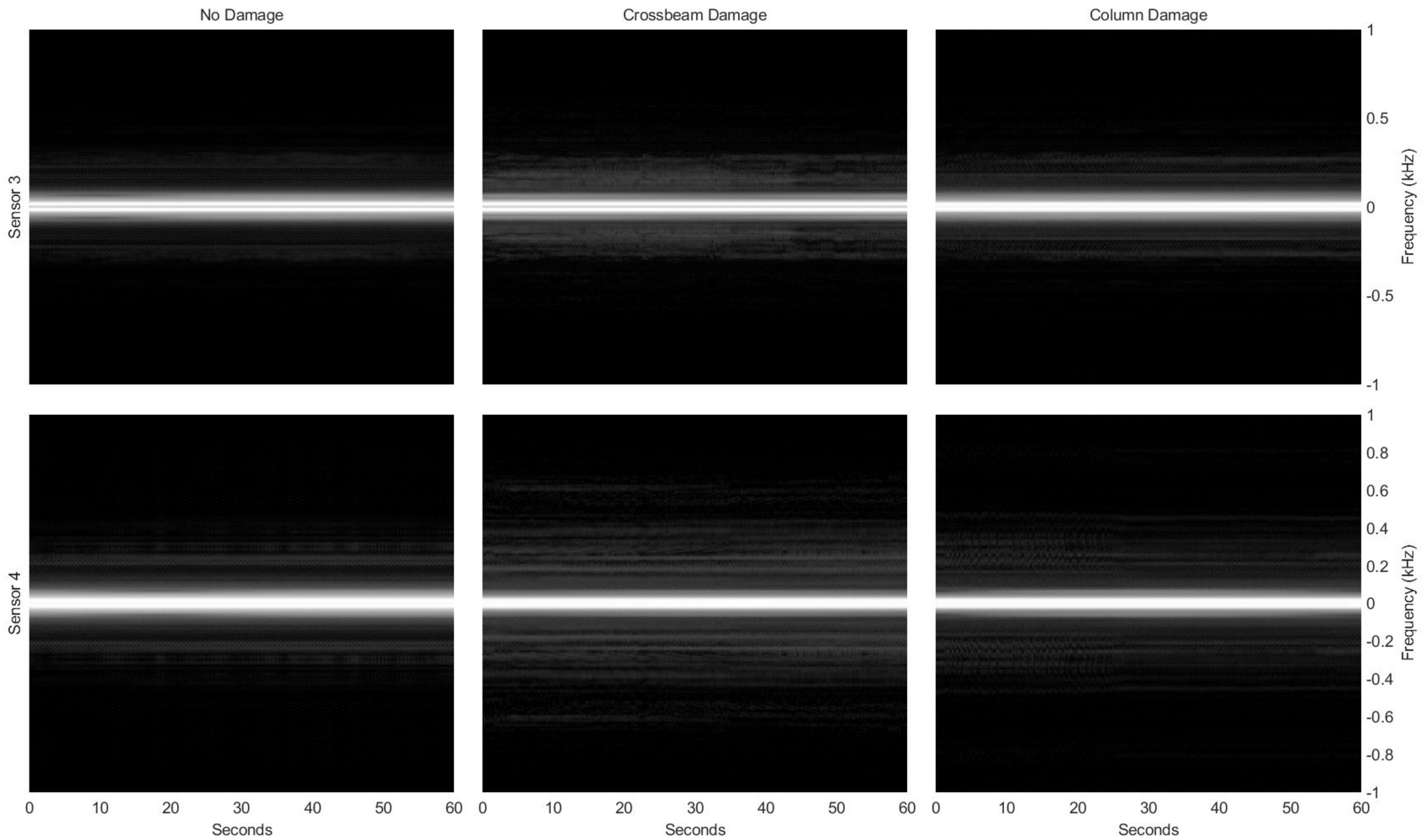


STFT 2-D plot



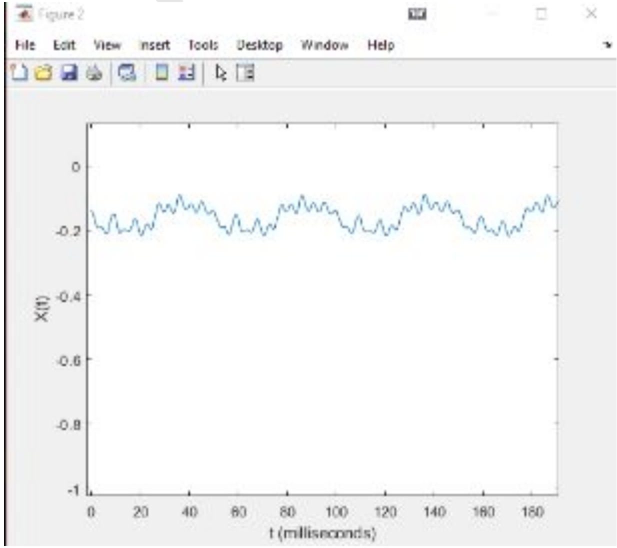
STFT 3-D plot


No Damage
Crossbeam Damage
Column Damage
Sensor 1
Sensor 2
Frequency (kHz)
Frequency (kHz)
1
0.5
0
-0.5
-1
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0 10 20 30 40 50 60
Seconds
Seconds
Seconds

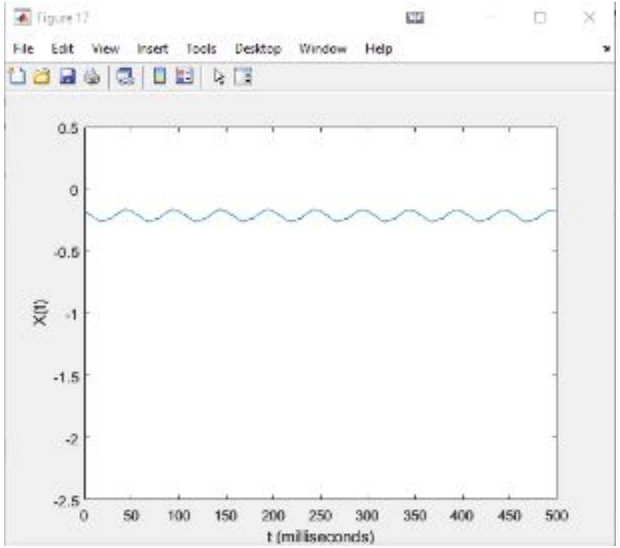


No Damage
Crossbeam Damage
Column Damage
Sensor 3
Sensor 4
Frequency (kHz)
Frequency (kHz)
1
0.5
0
-0.5
-1
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0 10 20 30 40 50 60
0 10 20 30 40 50 60
0 10 20 30 40 50 60
Seconds
Seconds
Seconds

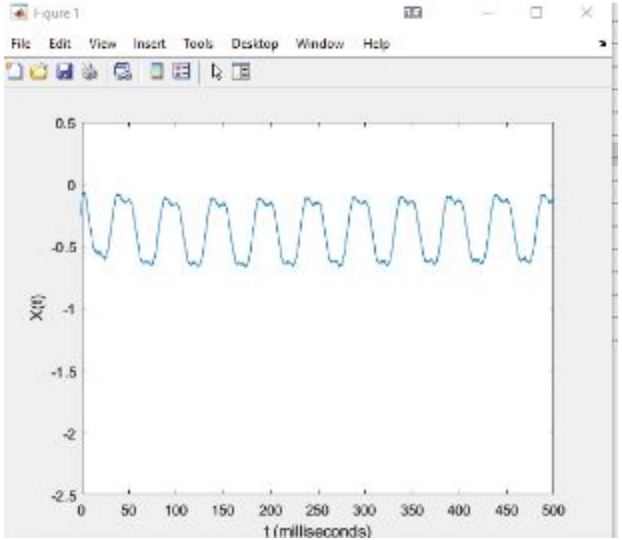
Appendix A: Preliminary Testing Data Results



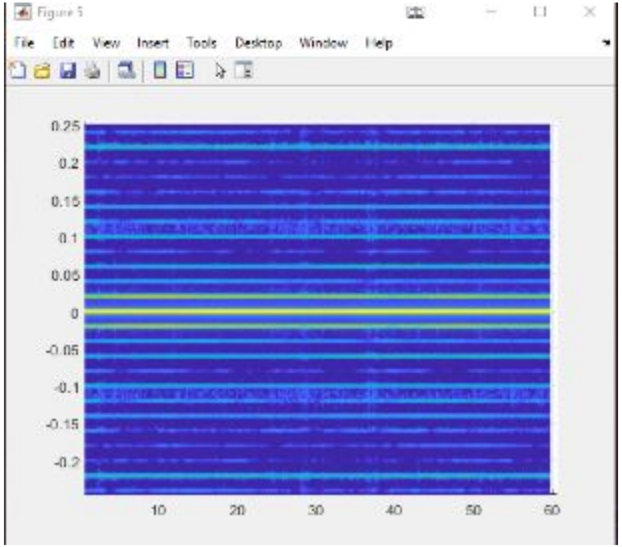
Undamaged Time Domain Data



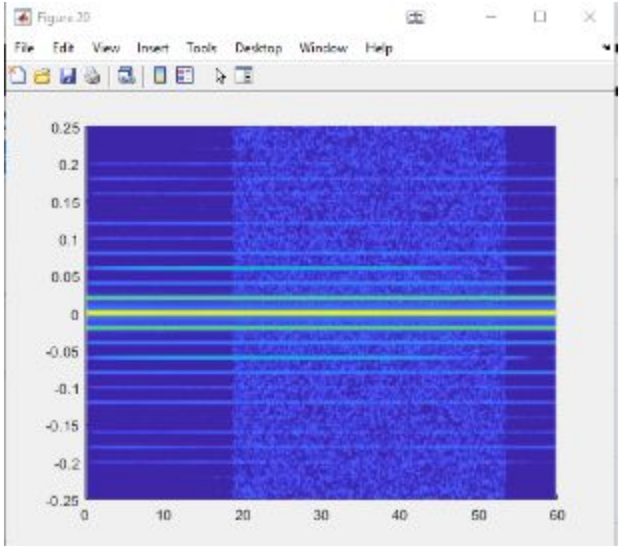
Partially Damaged Time Domain Data



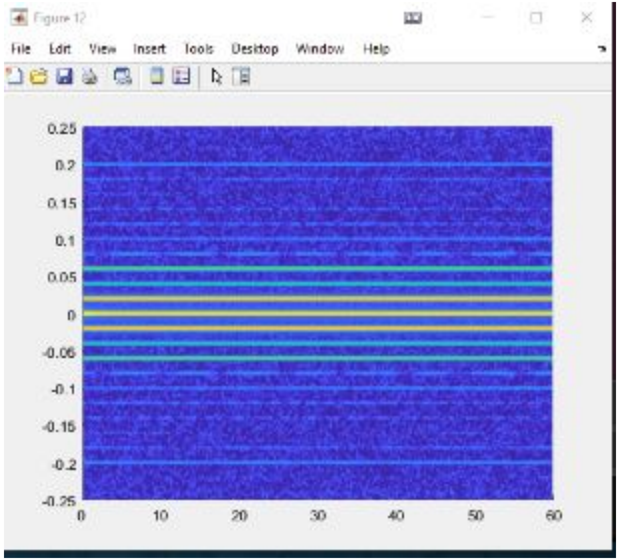
Fully Damaged Time Domain Data



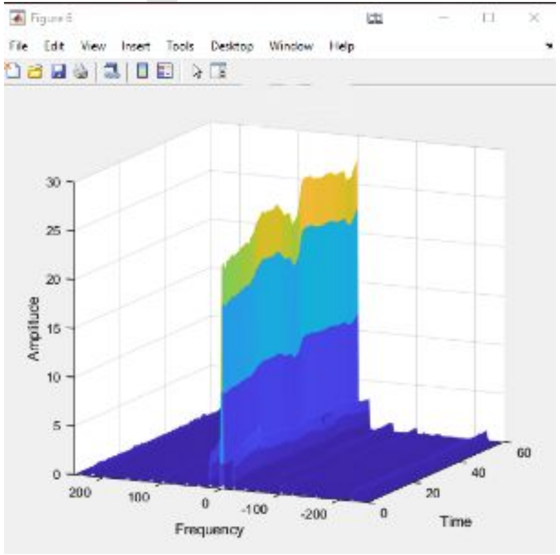
Undamaged 2D STFT



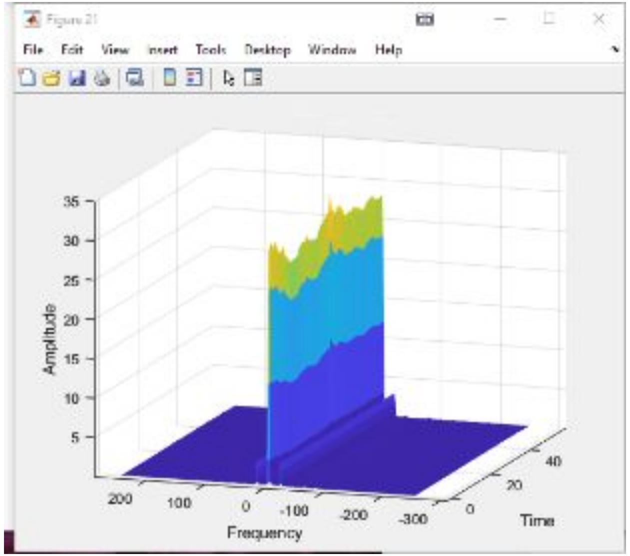
Partially Damaged 2D STFT



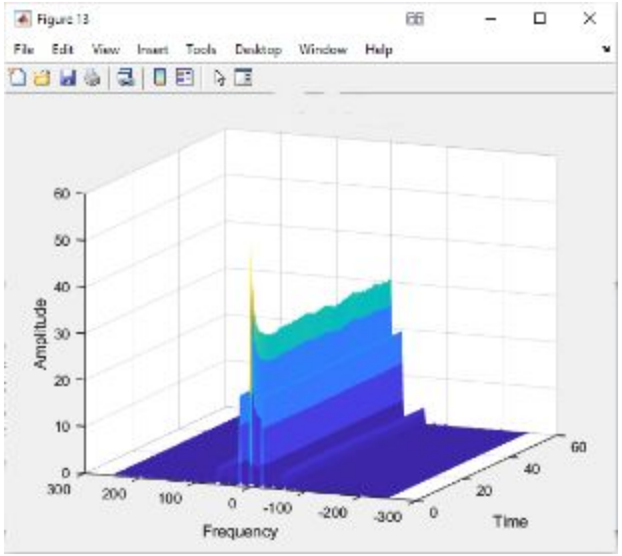
Fully Damaged 2D STFT



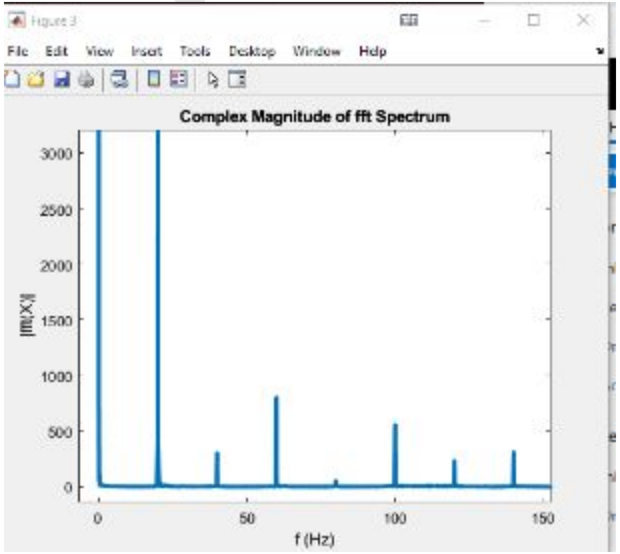
Undamaged 3D STFT



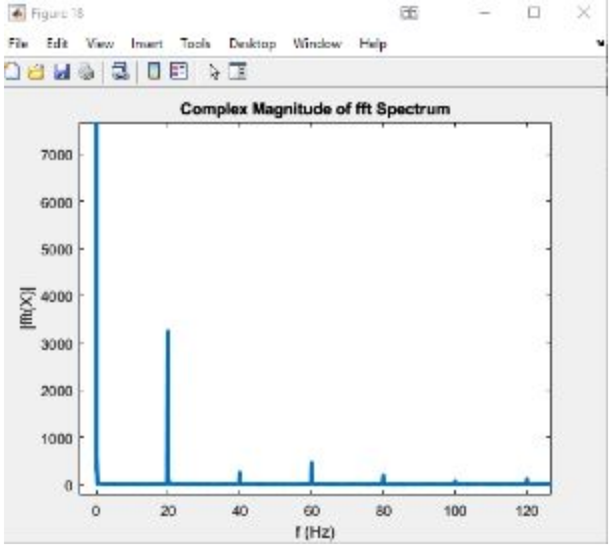
Partially Damaged 3D STFT



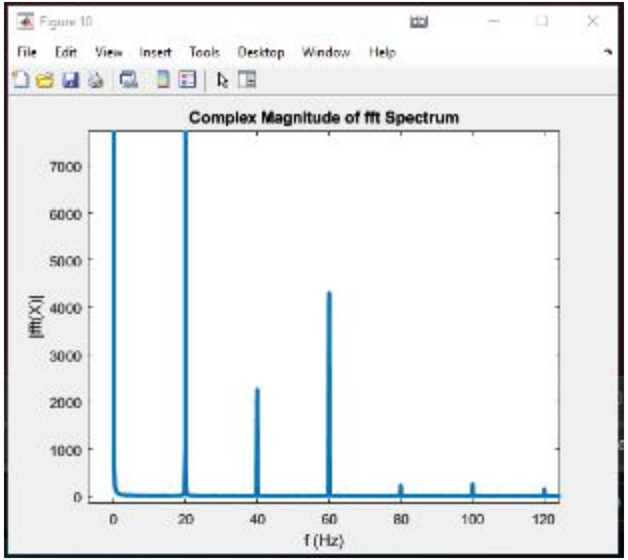
Fully Damaged 3D STFT



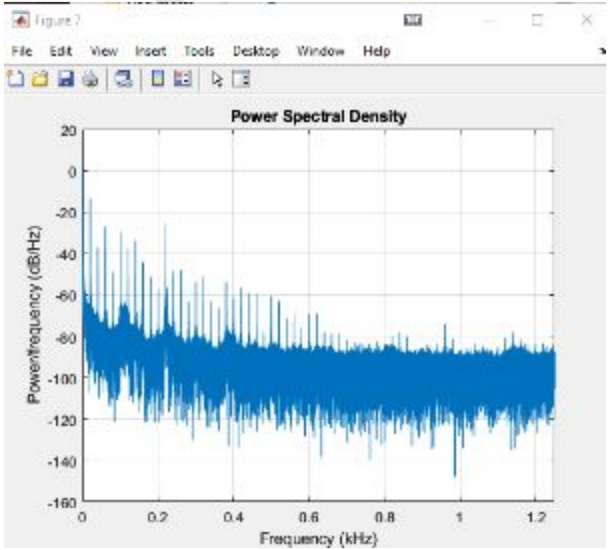
Undamaged FFT

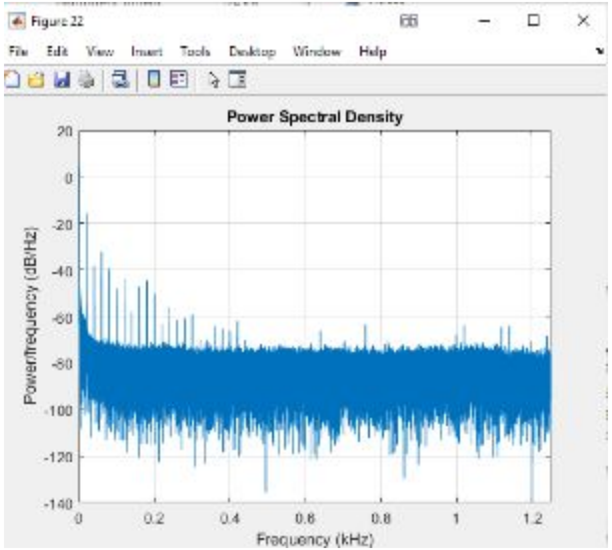


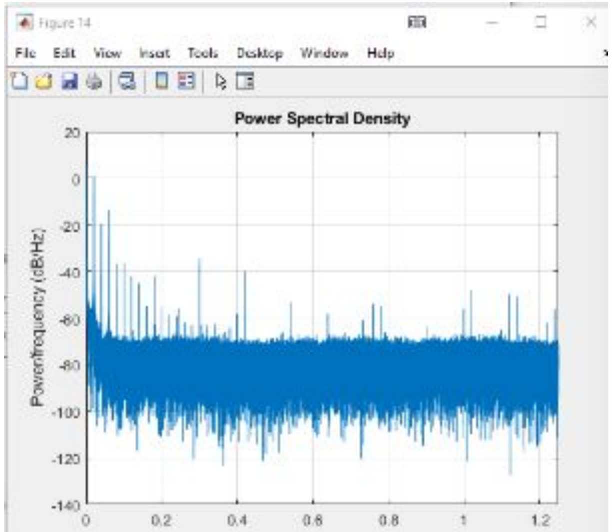
Partially Damaged FFT



Fully Damaged FFT



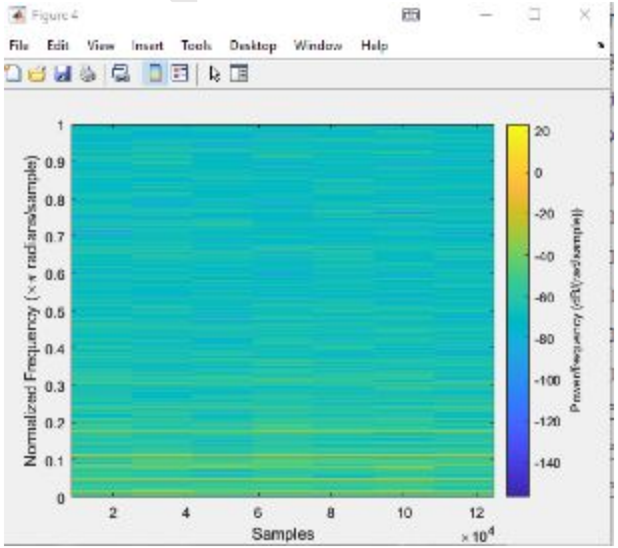




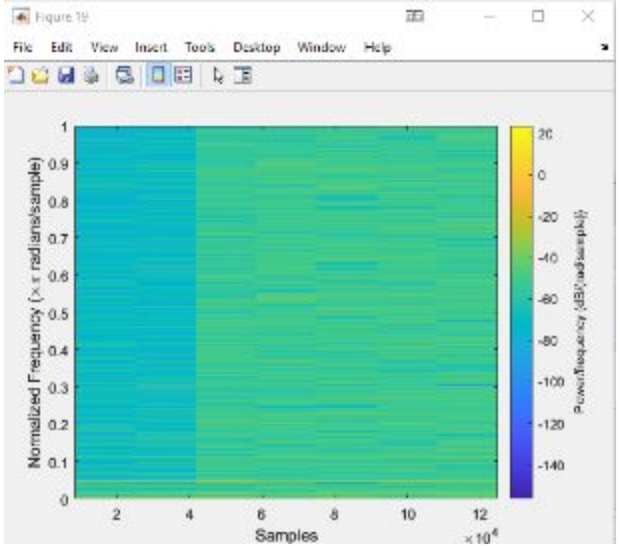
Undamaged
Power Spectral Density

Partially Damaged
Power Spectral Density

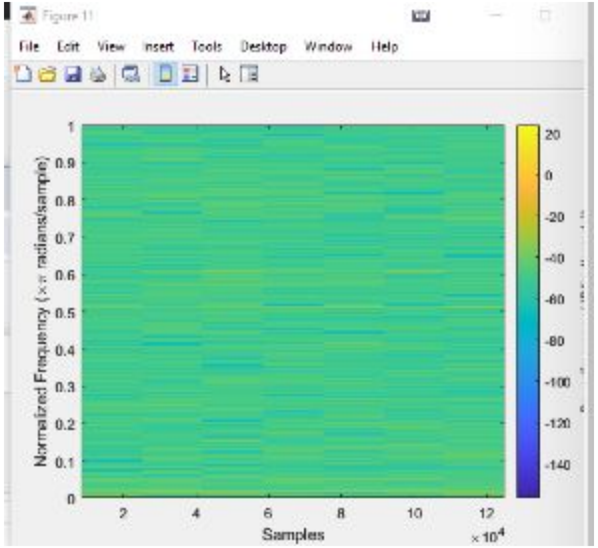
Fully Damaged
Power Spectral Density



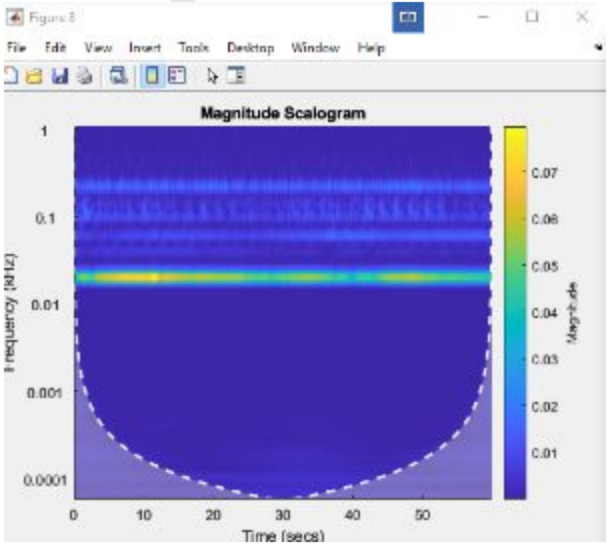
Undamaged
Spectrogram



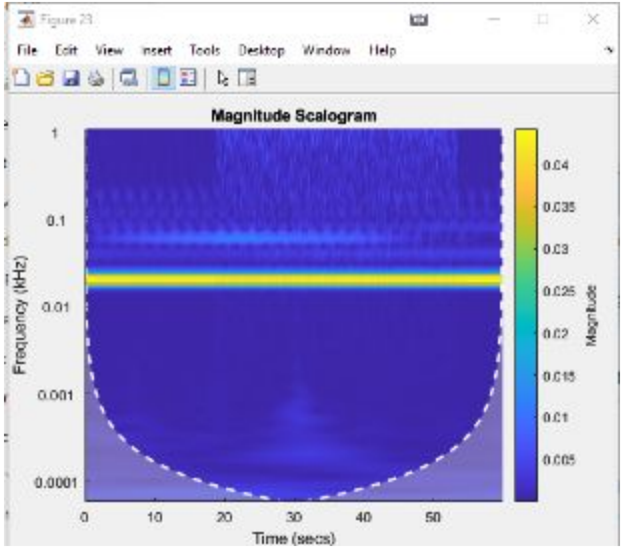
Partially Damaged
Spectrogram



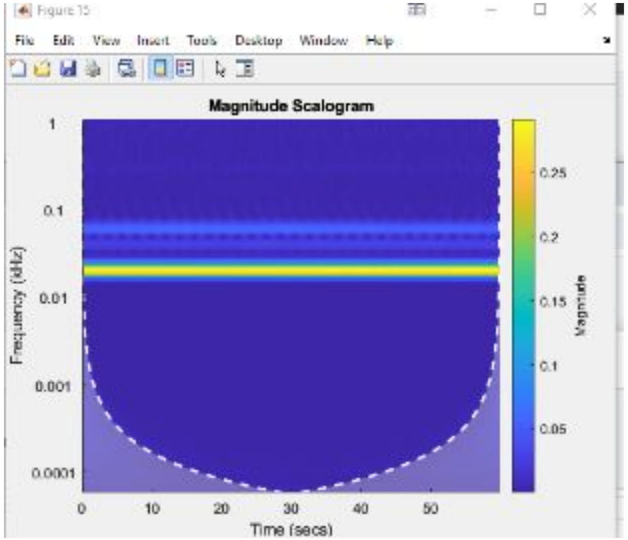
Fully Damaged
Spectrogram



Undamaged Scalogram



Partially Damaged Scalogram



Fully Damaged Scalogram