# *Air filter particulate loading detection using smartphone audio and optimized ensemble classification*

# Air Filter Particulate Loading Detection Using Smartphone Audio and Optimized Ensemble Classification

Joshua E. Siegel[a,b,*], Rahul Bhattacharyya, Sumeet Kumar, Sanjay E. Sarma

[a] *Field Intelligence Laboratory, Massachusetts Institute of Technology*
[b] *77 Massachusetts Avenue, Room 35-205, Cambridge MA USA 02139*

**Abstract**

Automotive engine intake filters ensure clean air delivery to the engine, though over time these filters load with contaminants hindering free airflow. Today's open-loop approach to air filter maintenance has drivers replace elements at predetermined service intervals, causing costly and potentially harmful over- and under-replacement. The result is that many vehicles consistently operate with reduced power, increased fuel consumption, or excessive particulate-related wear which may harm the catalyst or damage machined engine surfaces.

We present a method of detecting filter contaminant loading from audio data collected by a smartphone and a stand microphone. Our machine learning approach to filter supervision uses Mel-Cepstrum, Fourier and Wavelet features as input into a classification model and applies feature ranking to select the best-differentiating features. We demonstrate the robustness of our technique by showing its efficacy for two vehicle types and different microphones, finding a best result of 79.7% accuracy when classifying a filter into three loading states.

Refinements to this technique will help drivers supervise their filters and aid in optimally timing their replacement. This will result in an improvement in vehicle performance, efficiency, and reliability, while reducing the cost of maintenance to vehicle owners.

*Keywords:* Data mining and knowledge discovery, machine learning, emerging

*Corresponding author
*Email address:* j_siegel@mit.edu (Joshua E. Siegel)

applications and technology, intelligent vehicles, ambient intelligence

## 1. Introduction

Every year, the average vehicle's age and annual miles traveled increase[1, 2] and with the shift toward shared mobility, the need for efficient, reliable and durable vehicles continues to grow.

Most of the 260-million vehicle U.S. light-duty fleet [3] is gasoline powered [4], with engines that consume air and fuel, ignite this mix to propel a piston, and exhaust combustion byproducts. Any inefficiency causes engine performance, economy and longevity to suffer.

The intake system is critical to optimal performance. Incoming air must be free-flowing to attain efficiency, clean, to protect engine surfaces against abrasion and cold, so that the increased density allows more fuel to be combusted, improving power.

A key element of engine intakes, filters reduce contaminant concentration to safe levels [5] while ensuring free fluid flow to limit intake air heating. These filters are wear items, needing cleaning or replacement once loaded with dirt, dust, and debris.

Optimal filtration improves particulate entrapment, reducing engine cylinder erosion. Small changes to efficiency have significant impact: engine wear is 8 times faster for a filter that is 98% versus 99% efficient[5]. Further, ideal filtration reduces cabin noise levels and improves engine power and response. In contrast, dirty filters limit power, cause noise, waste fuel[6] [7] and may cause downstream catalytic converter failures. These challenges are most significant in carbureted vehicles[8] lacking closed-loop fuel control. While new cars switched to fuel injection by the mid-1990's, many cars, motorcycles and other light transport vehicles around the world still use carburetors.

Changing filter elements early seems an obvious solution, but early replacement causes subtle but serious problems. Particulate capture efficiency increases with loading[6], so lightly used elements reduce engine wear and extend service

life. There exists an optimal window in which to change a filter – one in which the filter captures a majority of particulates and minimally restricts flow.

In-vehicle sensors have been designed to solve the problem of optimal filter replacement, but most new vehicles with On-Board Diagnostics typically do not monitor this condition and older vehicles typically lack any sensing. Few vehicles offer vacuum-based intake pressure drop sensors [6] that indicate an increase over baseline pressure drop of 1-2.5kPa[7, 8]. Where sensors are not present, drivers are typically unaware and therefore rely on data-blind timing, with most drivers replacing filters at set intervals (often $15,000$km [7]) or when they look dirty. These methods are inaccurate, with vehicles used in varied environments with different particulate loads and unpredictable airflow rates[5].

In a survey of 21 air filters tested after removal, 15 were removed early while two had been changed after performance-degrading occlusion begun[7]. This indicates that drivers taking vehicles in for service change filters too early but is inherently biased, as the dirtiest filters are found in those cars never taken for service. Assuming a 2% loss in fuel economy in the under-serviced vehicles, an average driver spending $\$1,680$ per year in fuel wastes \$33.60 driving with a dirty filter. This exceeds a typical filter's cost and demonstrates the potential savings for optimal replacement timing, not to mention the long term damage to engines and catalytic convertors.

Streamlined, realtime filter classification could reduce vehicle operating costs and emissions while improving reliability. There exists latent demand for this information – 81.4% of people would take recommendations from a data-informed system [9].

To reduce the need for behavioral changes, low-cost, pervasive sensing using smartphones may be used to repurpose existing devices[10]. In recent years, consumer electronics manufacturers have increased mobile sensing capabilities. These new inputs, ranging from atmospheric pressure and device orientation to temperature, touch, and proximity, have met with commensurate enhancements in mobile computation, storage, and connectivity [11]. Our own work has shown that it is possible to monitor engine ignition using such devices [12].

We aim to transition from today's reactive maintenance paradigm to proactive, availing ourselves of these resources. We apply mobile audio to observe how a car "breathes" to classify air filter performance with the goal of creating a "remaining life" indicator and condition monitor for air filters to enhance compliance with automotive best maintenance practices. This paper demonstrates how mobile audio data and ensemble classification may be applied to categorizing air filter condition into multiple loaded states.

In Section 2, we hypothesize that sound emanating from the intake changes with particulate loading, while Section 3 explores related work. Section 4 describes an experimental procedure to collect data and simulate contaminants restricting airflow. We describe our ensemble classification algorithm in Section 5 and present results in Section 6, showing high accuracy in differentiating new, gently used, dirty and obstructed filters. Finally, Section 7 discusses of future improvements for this algorithm and applications of pervasive sensing to other vehicle faults.

## 2. Problem Description

The ideal combustion engine demands a limitless supply of free-flowing, clean, cold air. In reality, engines require filters to clean air and limit wear. When new, these filters restrict intake airflow, and as the filter loads with contaminants, this restriction and related pressure drop increase. While intake systems are tuned to minimize noise, vibration and harshness, changes in flow ultimately lead to perceptible changes in the audio emanating from the intake. We assert that these pressure and flow changes related to occlusion may be discerned using digital audio samples to inform machine learning algorithms.

In intakes with a new filter, the housing is the primary cause of pressure drop. With a loaded filter, the system's pressure drop is dominated by the filter element[5]. Since the filter housing is rigid, it's pressure drop remains largely constant irrespective of filter loading. This suggests that as filter loading increases, the pressure drop will bias from housing to filter. Experimental

analysis corroborates this trend[6].

We listened to an idling vehicle with old and new filters and hypothesized that contaminant-based blockage leads to an increase in audio frequency as more air passes through smaller filter apertures. This is because modern engine control systems hold volumetric airflow constant, even as filter loading changes.

We further note that flow rate reaches an asymptotic limit as loading increases. In Jaroszcyk's Figure 4 [5], high flow rate filtration efficiency plateaus at a critical loading point due to particle reentrainment, suggesting there may be a similar plateau indicative of an optimal changeover point where the filter is at peak efficiency and does not yet limit flow.

Two factors complicate audio-based particulate identification: the intake is tuned to attenuate noise emissions to minimize occupant annoyance[13, 5], and engine intakes vary across vehicles.

As a result of this first issue, intake noise is not clearly audible in the car's cabin – and though the air intake system's sound may be distinguished under the hood of an operating vehicle, changes in filter loading contribute small signals to large, combustion-related noise. To maximize classification accuracy, we will record audio under the hood to reduce background contributions. Further, we record one data set from a stationary microphone to minimize variability and another from a moving microphone representative of how a person might use a phone to record samples.

To resolve intake geometry variance, we propose creating new models for different vehicle types. We therefore will collect data and train models for two distinct vehicle types which might be used to classify vehicles with similar engines. These "pseudo-custom" classifiers will ultimately allow mobile devices to identify the optimal inflection point between clean and dirty filters.

## 3. Prior Art

Characterizing vehicle performance, classifying component condition and identifying abnormal behavior using time-domain signals is not a new field. In-

vehicle sensing in particular has been applied to air filter monitoring. However, multi-state classification and pervasive condition monitoring remain underexplored.

This is not for lack of pervasive vehicle diagnostics. The use of audio signals is especially prevalent in research and industrial applications because acoustic signals do not require rigid contact between a sensor and the instrumented object.

Researchers have used audio processing to identify cylinder misfire faults, capturing the sound from an engine using a microphone and analyzing the signal in the time and frequency domain to differentiate normal and abnormal engine operation[12]. Dandare [14] and Sujono [15] used dedicated recording equipment to classify engine misfires in a laboratory environment, with Dandare's artificial neural network's accuracy ranging from 85-95%[14]. Kabiri and Ghaderi [16][17] used noisy samples with principal component analysis and correlation-based feature detection to obtain accuracies of 70-85% in identifying misfires.

Beyond identifying misfires, Wu et al's audio-based approach uses the discrete wavelet transform and Parseval's theorem to decompose signals into constituent energy distributions as input into a neural network capable of identifying air intake manifold leaks, ECT and camshaft sensor failures, and cylinders with accuracies exceeding 95% accuracy[18]. Yadav et al. use audio feature signatures to compare samples against known-good or known-failed sample data, conducting Sumpeak analysis using a band-pass filter to window input into a critical frequency range and applying the Hilbert transform. This approach uses cross-correlation to detect whether an engine is normally or abnormally operating and subsequently conducts secondary correlation analysis on faulty engines to further differentiate faults with up to 91% normal/abnormal detection accuracy[19]. Kemalkar applied similar concepts to diagnosing motorbike combustion engines, using mel cepstrum and Fourier features to identify normal versus abnormal operation and then conducting additional classification to differentiate faults, with accuracy ranging from 52-79% on faults related to engine oil, piston rings, or valve timing[20].

Outside of applications for audio data to identifying vehicle faults and maintenance needs, other pervasive signals such as GPS-derived location histories or accelerometer-inferred vibration patterns have been used to identify suspension-related malfunctions[21, 22]. Vibration analysis can also determine cylinder head faults[23].

Contemporary studies focus on classifying vehicle condition into binary states of normal or abnormal operation, whereas condition identification and remaining life calculation would yield actionable insight to reduce a vehicle's total cost of operation and improve longevity. We propose to conduct condition-based monitoring to identify air filter maintenance needs before a fault presents.

Exploring domains such as optimal, data-informed air filter replacement timing will ultimately improve vehicle efficiency, performance, and reliability similarly to how on-board sensors can be used to indirectly monitor the wear of engine oil to optimize change intervals[24].

## 4. Experimental Procedure and Hypothesis Validation

This section describes how we generated audio samples from a vehicle with varying degrees of air filter contamination to train a three-state classifier for multiple vehicles.

We first discuss an experiment collecting data to prove the concept of using audio features to differentiate old from new filters. Then, we present a procedure for generating controlled data for multi-state contamination classification and explain how this approach assures our classifiers' robustness. Here, we collect data from two cars using stationary and mobile microphones to create labeled training and testing data. The result is three labeled data sets: one from a Honda Civic and a stationary microphone, one from a Honda Civic with a moving iPhone, and one from a Mazda 2 with a moving iPhone. Finally, we describe an experiment simulating non-uniform filter occlusion such as is caused by entrapment of a leaf in the intake airbox. These experiments generate data useful to identify the loading state as well as the type of contaminant.

### *4.1. Uncontrolled Data Collection*

Our initial experiment used a 2014 Mazda 2 with a normally aspirated inline four engine. This non-turbocharged engine provided clearer audio signal relative to a forced induction variant.

We allowed the vehicle to warm up to operating temperature by driving for five minutes, ensuring the use of a consistent engine calibration table across runs and eliminating the presence of a "fast idle" signal contribution. Maintaining constant engine temperature is important to control for air intake density and flow rate and the resulting audio changes.

Once at operating temperature (180F in this case), the vehicle was put into park and allowed to sit for one minute at idle to allow the transmission's rotating elements to spin down. The engine was left operating at idle for data collection to minimize the amplitude of the combustion audio relative to the engine's "breathing" noise. This noise comes from the intake orifice and is impacted by engine load and speed[13], so testing at idle minimizes both variables' impact. We recorded samples from under the hood to minimize the impact of sound deadener and to increase the signal-to-noise ratio.

This process closely mirrors our process for collecting misfire audio[12], with the user recording data for each vehicle and every test case with the phone constantly moving under the hood. This approach allowed us to collect data from a phone with changing orientation and different reflected and environmental noise. This approach helps create training data that are location and orientation insensitive. This ensures that our classifier's features (discussed in Section 5) are robust to physical perturbation.

We recorded binary classification data first, recording samples for new and dirty air filters. We first used a new filter to ensure a clean baseline. The filter is shown in Fig. 1

We recorded ten minutes of audio data from the new filter onto an iPhone 6S, though in our frequency range of interest (20Hz-20kHz) most smartphone microphones perform similarly. The widely-used MEMS microphones in mobile

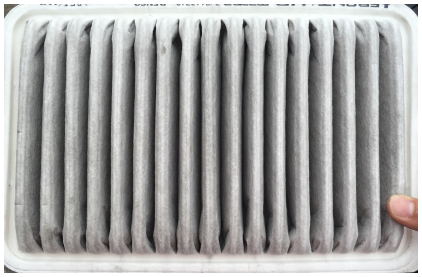

Figure 1: The new filter is light in color and the areas between pleats are clean.

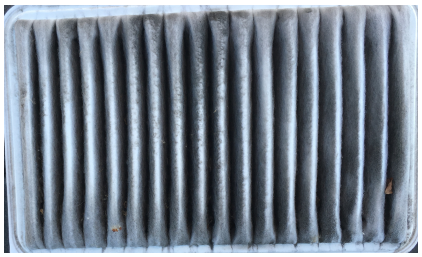

Figure 2: A dirty filter shows significant particulate build-up, causing the element to discolor. Areas between pleats show significant debris.

phones have flat frequency response across their operating range, so power-normalized data should be interchangeable for classification regardless of the originating device. The volume of the engine is similar to the volume of human speech, so smartphone microphones will have an appropriate sound pressure level to minimize distortion and saturation. These behavior similarities have been explored on enthusiast websites. [1]

Data were recorded to an uncompressed stereo WAV file at 48kHz. The iPhone was moved and reoriented across the entirety of the engine's visible surface. An example showing the phone in a representative orientation and distance from the engine is shown in Figure 3.

We then replaced the clean filter with the dirty filter (c.f Fig. 2) that had been present in the Mazda 2 and repeated the process. There was no way to quantify what operating conditions had contributed to the dirty state of the Mazda 2's filter as the filter was obtained from a rental vehicle, but we believe the filter to have approximately $30,000$ miles of contamination on it based on the vehicle's mileage and visual inspection.

[1] `http://blog.faberacoustical.com/2010/ios/iphone/iphone-4-audio-and-frequency-response-limitations/`

### *4.2. Validating Binary Classification*

We conducted frequency analysis for the clean and dirty filter data, first dividing the minimum 10.5 minutes (630s) of data into 252 chunks of 2.5s each. For each chunk, we normalized the data by the root mean square (RMS) value to account for variations in phone position and environmental noise. We then computed a Fourier Transform (FT) of the time domain data for each chunk. We binned the FT transform data into 1Hz bins and considered spectral components between 20 Hz and 20 kHz. We assigned a lower cutoff of 20 Hz to avoid contamination by $\frac{1}{f}$ noise. We imposed the upper cutoff of 20 kHz as it is the upper limit for human auditory perception. We did this because we cannot be confident that consumer mobile device microphones perform well outside this range, and therefore avoided sampling right up to the Nyquist cutoff frequency of 24 kHz. Finally, we averaged each of the 252 FT and plotted them for visual comparison. Fig 6 shows noticeably different energy content between the clean and dirty filter and lends credibility to our idea that particulate loading generates a different auditory signal.

### *4.3. Controlled Experiment, Multi-State Classification*

Following visual validation, we needed data to prove robustness for multiple vehicles and states, as well as to determine whether moving microphone data works reliably. The following section describes the experimental procedure and data collection process we used to collect three labeled data sets: one set from a Honda Civic recorded with a stationary microphone, one set from a Honda Civic recorded by a moving iPhone, and one set from a Mazda 2 recorded by a moving iPhone.

We planned to record samples from a moving iPhone 6S to simulate how a "point and click" app might work. For the stationary microphone, we set up a TASCAM DR-40 recorder connected to a stationary microphone located approximately 30cm from the air intake. This is shown in Fig 4.

It takes thousands of driven miles for a vehicle's filter to transition from clean to filthy. For the purposes of our experiment, we wished to reliably simulate the

Figure 3: This photo shows the phone being moved across the engine's surface while recording audio. Relative motion is more representative of real-world use, e.g. when the algorithm is running self-contained as a mobile application.

Figure 4: This figure shows a stand microphone placed over the air intake, to record flow data. The stationary microphone and proximity to the air intake minimized the impact of extraneous engine noise on the recording.

transition on demand. We therefore removed the air filter and applied a layer of carbon filter material approximately 2mm thick to the filter's exterior (intake facing) side to represent uniform particulate buildup. This additional material was applied tightly to the base filter to minimize leakage around the filter. This is shown in Fig 5.

We then collected 10 minutes of audio data using the process from Section 4. We first collected data with the filter covered with carbon filter material and obtained the average FT profile as discussed in Section 4.2 to determine whether this experimental procedure is representative of real-world filter loading. Fig 7 shows the response of simulated particulate build up. A comparison with Fig 6 confirmed that the carbon filter occlusion closely approximates particulate build up and therefore is an appropriate surrogate for data collection.

To simulate additional contamination, we repeated the process of adding carbon filter layers three more times. We collected data from a new filter and each variation from one to four layers of additional filtration material. This resulted in a total of five sets of training data, ranging from a brand new filter to

Figure 5: Simulation of filter loading using carbon filter material. Additional layers provide additional restriction, representing more significant occlusion.

a filter with four layers of carbon paper providing additional obstruction.[2] Fig 8 illustrates the average FT response of all five states. There are discernible peaks for all five states that a classification algorithm could use for state determination. However, there also appears to be the potential for state confusion in the two highlighted zones shown in Fig 9. In these zones, the amplitude of the clean and one layer case is very similar. The amplitude of the three and four layer case is similarly likely to be misclassified.

Based on this confusion and our hypothesis that the air filter reaches a form of asymptotic loading from Section 2, we chose to reduce our state space and focus on classifying only states with a new filter, one restrictive layer, and four layers of material.

We repeated this data collection process for both a Honda Civic and Mazda 2 with normally-aspirated, four cylinder engines. Section 5 discusses the accuracy with which we can differentiate among these three states using our optimized classifier. We examine models for the Honda with the iPhone and stand micro-

[2]These samples are available for download on Harvard Dataverse: `https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/3PDXSI`.

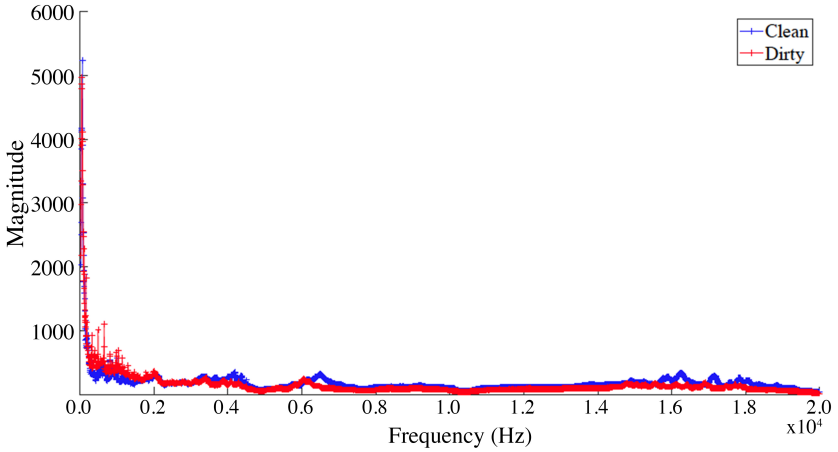


Figure 6: A comparison of the FT response of a clean and dirty filter. There are noticeable differences in the FT peaks for the two filters.

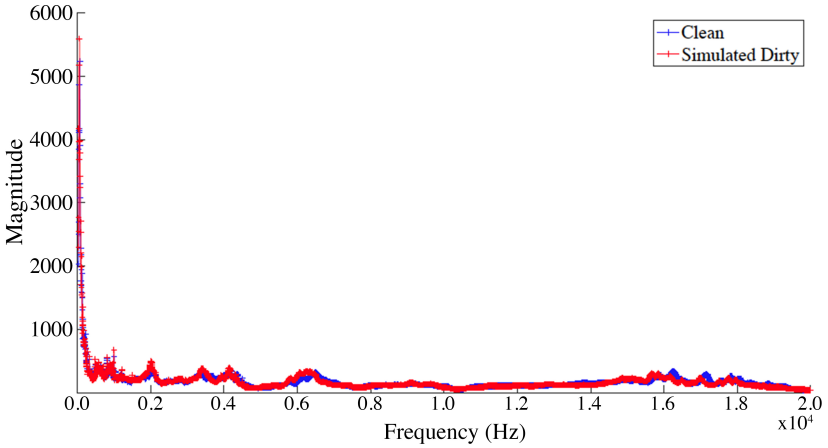


Figure 7: A comparison of the FT of a clean filter and a filter covered with 1 layer of carbon filter material. The position of the differences in FT peak positions resemble actual particulate buildup.

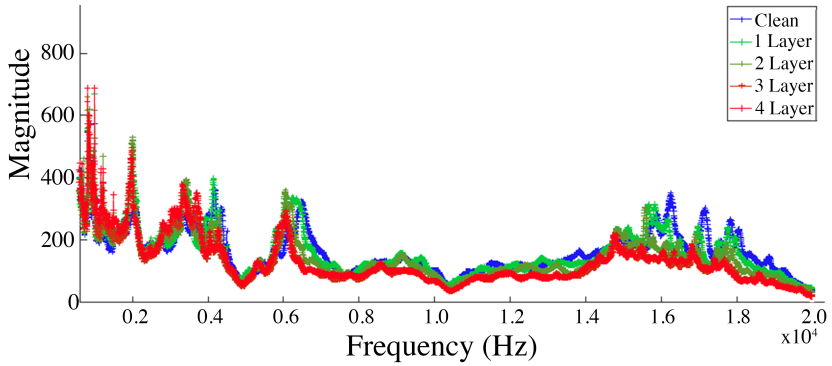


Figure 8: A comparison of the FT response for a clean filter and filters with 1-4 layers of attached additional carbon filtration material. The figure focuses on a frequency range where the peak differences are more noticeable.

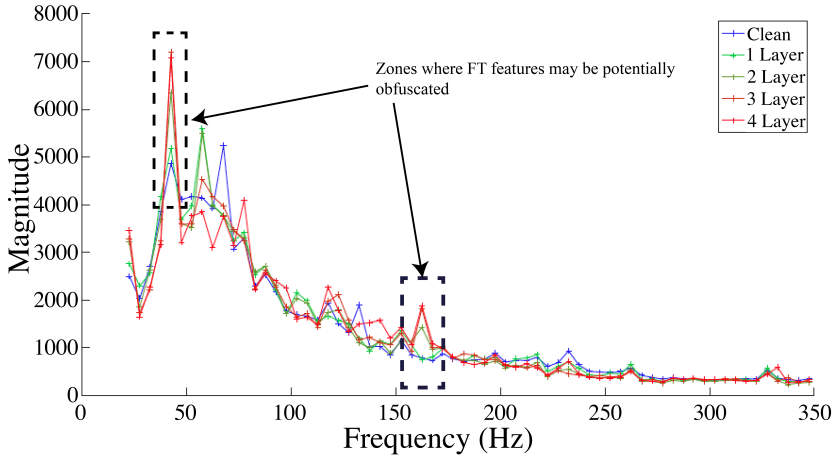


Figure 9: Zones of potential feature obfuscation for the clean and 1-4 layers of carbon filter material. The frequency axis has been truncated to 350 Hz so that the individual peaks can be easily discerned.

phone data, and Mazda with the iPhone.

### *4.4. Controlled Experiment, Large Occlusion*

Frequently, large debris such as leafs enter engine intakes. These occlusions may become stuck to the air filter in operation and cause lean engine operation. This can decrease power and increase engine operating temperatures and thus should be avoided.

As a final experiment, we removed all of the carbon filter layers and collected a set of data testing for the presence of large, nonuniform contaminants by inserting a mid-size (10 cm $\times$ 10 cm) piece of paper inside the airbox on the exterior side of the filter (c.f Fig 10). This simulated the presence of a leaf or other foreign body in the intake.

From these data, we observed a new peak forming in the Fourier Transformed data. This peak indicates that the occluding material creates a strong signal at a particular frequency, suggesting that the paper may be acting similar to a reed on a wind instrument. This bifurcated peak is shown in Figure 11. This feature suggests we will be able to differentiate filter loading between large contaminants and particulates using only audio.

## 5. Algorithm Development

From Fig 5, we hypothesized that the FT peak differences could be used to differentiate among three states, with additional features improving classification accuracy. In this section we discuss how we generated features, tuned a classifier, and selected the optimal input parameters to maximize filter loading classification accuracy while minimizing overfitting.

### *5.1. Feature Generation*

Using only Fourier features led to poor outsample results. Based on our previous successful diagnostics[12], we therefore additionally generated mel-frequency cepstrum and wavelet features.

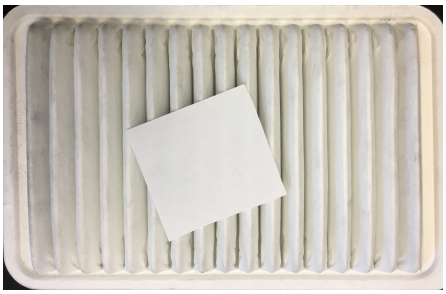

Figure 10: Use of a piece of paper to non-uniformly constrict the filter to represent discrete foreign body loading, such as occurs when a leaf enters the intake system.

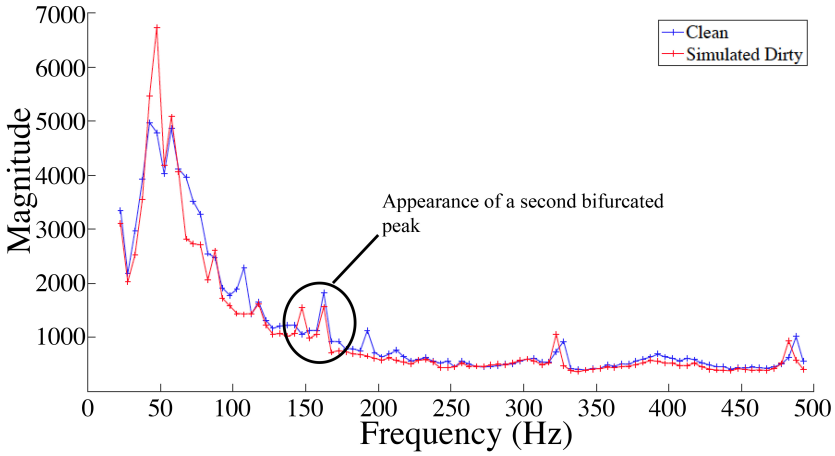


Figure 11: Appearance of bifurcated peak for filter non-uniformly covered with a piece of paper. The frequency axis has been truncated to 500 Hz so that the individual peaks can be easily discerned.

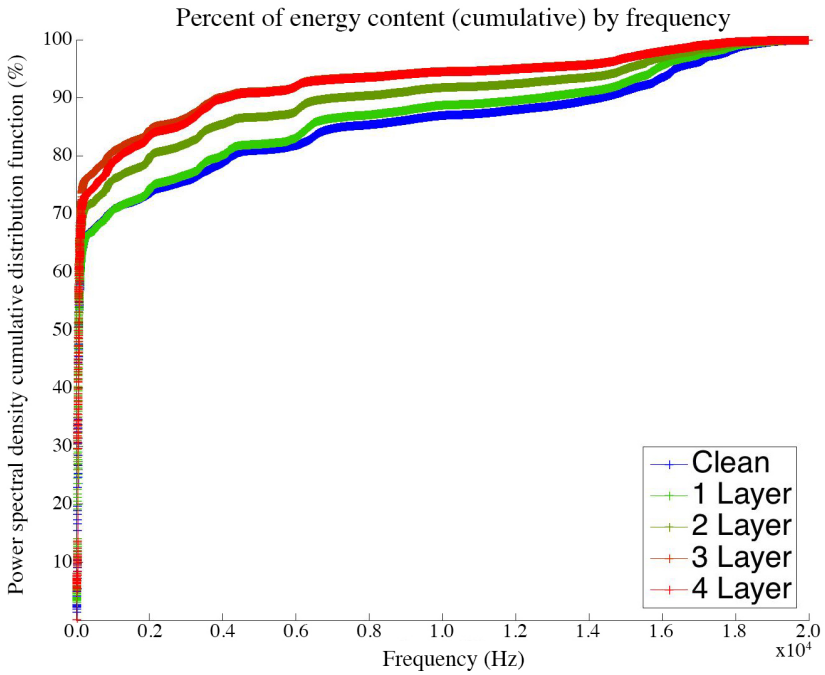


Figure 12: This plot shows the cumulative energy distribution function for the Mazda's full five states. We used this cumulative energy function to determine an appropriate cutoff frequency for input to the classifier.

To reduce computation time, we windowed the DFT results, selecting 20 Hz as a cutoff limit based on typical smartphone microphones, and examined energy content to set the appropriate upper cutoff limit. We computed the cumulative energy contribution versus frequency for each input set and found the 75% cutoff to be near $3,350$ Hz, striking a balance between computation time and feature richness. The CDF is shown in Fig 12.

Whereas the Fourier Transform decomposes a signal based on a model of sinusoidal waves, the Wavelet Transform decomposes a signal based on functions in the Fourier space as well as the real space. We used the Discrete Wavelet Transform (DWT) at level 10 using Daubechies 4 wavelet to calculate 33 features from each input segment, including the mean, standard deviation, and skewness at each level of signal decomposition.

Mel-Frequency Cepstral Coefficients use short-term frames from an original signal to create a spectral signature useful for classification. We applied a frame size of 1024 with each frame incrementally shifted by 512 samples, extracting 12 coefficients from each frame. We used the GNU "voicebox" toolbox[3].

We concatenated these DFT, DWT, and MFCC features to form training and testing vectors for each labeled set.

### *5.2. Classifiers*

Machine learning algorithms aim to prove a hypothesis, using high dimensional inputs and limited training data. It is a challenge to avoid overfitting while learning a stable classifier capable of making predictions on unseen data. Ensemble learning is an approach where several models are learned and the final prediction is made by a weighted or unweighted vote of the individual classifiers to improve robustness[25].

While we preferred Adaboost based on our misfire work[12], this approach often fails in multi-class learning[26] . We therefore decided optimize our performance and robustness using bagged trees based on Breiman's random forests[27].

Bagging (bootstrap aggregating) is an ensemble learning method that works by training several classifiers on a random sampling (with replacement) of the original training data. The bagged classifier is then the majority vote of the individual classifiers. Bagging increases the stability of predictions while reducing variance and overfitting and has been successfully applied in bioinformatics[28], finance[29], sensor networks[30] and health care[31].

### *5.3. Grid Search*

Upon selecting a bagged classifier, we developed an approach for model training, testing, and validation. We first decided to allow 25% leave-out data for model testing and used the remaining 75% for 5-fold cross-validated training

[3]Voicebox is GNU licensed; available `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`

and testing as we conducted a grid search to identify each cars' best-performing hyperparameters.

To find the optimal model, we varied parameters including:

- **Time Segment Length (s):** 0.5, 1, 2, 5
- **DFT Bin Size (Hz):** 0.25, 0.5, 0.75, 1, 2, 3, 4 and 5
- **Number of Trees:** 10, 100 and 1000
- **Maximum Tree Splits:** 5, 10, 15 and 25

Subtle changes could significantly alter the classifier's performance: increasing the length of each time segment used for feature generation decreased the number of clips used for training, testing, and validation, but increased immunity to small perturbations in input signal. Increasing the DFT bin size allowed more features to be lumped together, reducing the number of DFT elements but making it harder to discern narrow bandwidth features.

For each parameter, we determined likely maximum and minimum values and grid sizes to bound search time. We limited the maximum segment time to $t = 10.0$s to ensure a sufficient number of samples for training and testing, and to ensure a reasonable maximum time for users to hold their mobile devices near their vehicles. To simplify computation, we imposed a minimum DFT bin size of 0.1 to limit the number of features generated, while we constrained the maximum bin width to 5.0Hz to capture narrow-band features. Finally, we capped the number of trees bagged at $1,000$ to assure that this application might be able to run quickly on typical desktop computers. The number of maximum splits allowed in each tree was selected to ensure that at the low end of 5 splits, most trees would undergo pruning, while the upper limit of 25 splits would likely exceed the splits present in unpruned trees.

For each data set and classifier permutation, we calculated the 5-fold cross-validated accuracy from insample data. For each car and recording setup, we selected the best validated result (determined as having the minimum misclassification rate/ maximum 5-fold classification accuracy on the insample data).

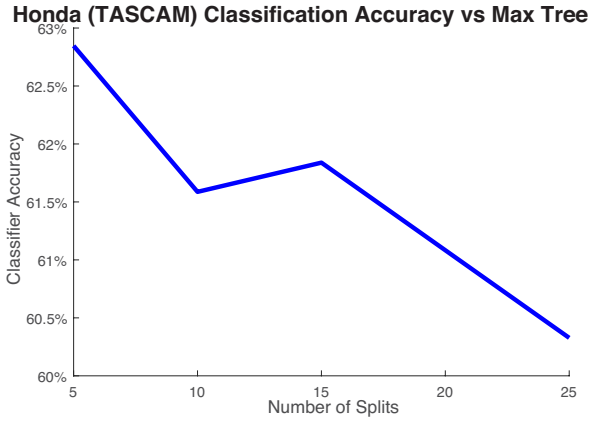


Figure 13: This plot shows how the algorithm's insample classification accuracy varies based on maximum number of splits allowed in the classifier's constituent trees. We see here that the 5 branch case works best, suggesting that reducing the number of splits may lead to the creation of more common decision points across the ensemble's trees.

To ensure that these results are stable, we generated a series of plots for each car showing how insample predictor performance varies with changing parameters. In each, we selected the optimal value for all parameters except that being permuted. Representative figures for the Honda with stationary microphone are shown in Fig 13-16.

Finally, we used these optimized tuning parameters to calculate the outsample performance using the 75% insample data for training and the remaining 25% hold-out data for testing.

### *5.4. Ranking*

Though filtering based methods can rank individual features, they often miss non-linear multivariate interactions[12]. Bagged trees, however, learn the importance of variables as a part of the model learning process. The importance score of a feature is calculated by computing a normalized measure of the reduction in classification error due to the feature's associated splits[32]. Therefore, we used MATLAB's "predictorImportance" parameter for ranking the features

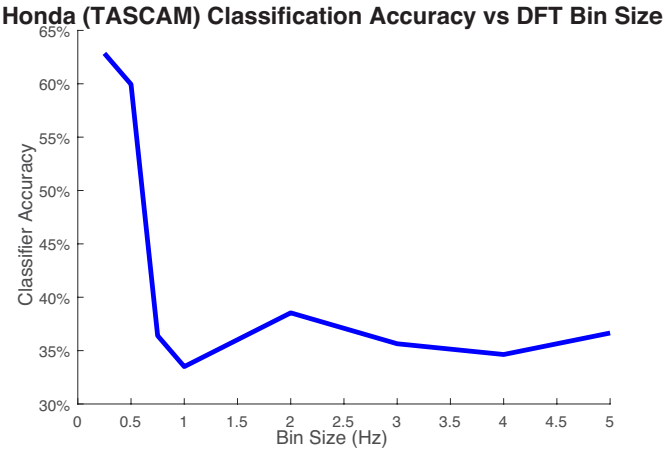


Figure 14: This plot shows how the algorithm's insample classification accuracy as it changes with different FFT frequency bin widths. We see a maximum at the low end of our allowable bin size, 0.25Hz, suggesting that the constituent trees are making use of narrow-band peaks for classification.

and eliminate all zero-importance features from the training set.

Reducing the feature count using predictor rank is important when considering implementing this approach on a mobile device, as these features no longer need to be generated, uploaded, or stored. The net result is faster computation, improved mobile battery life, reduced file sizes, and no impact on accuracy (as the classifier already ignored all zero-importance features).

## 6. Out-Sample Results

This section shows the outsample, optimized results for the three tested data sets in tabular form.

In Table 1, we see the optimal configuration for each of the three models and the 5-fold cross-validated insample performance as well as the 25% outsample performance.

We note trends in optimal model parameters. Each model tends to select small bin sizes, suggesting that the DFT elements will play an important role

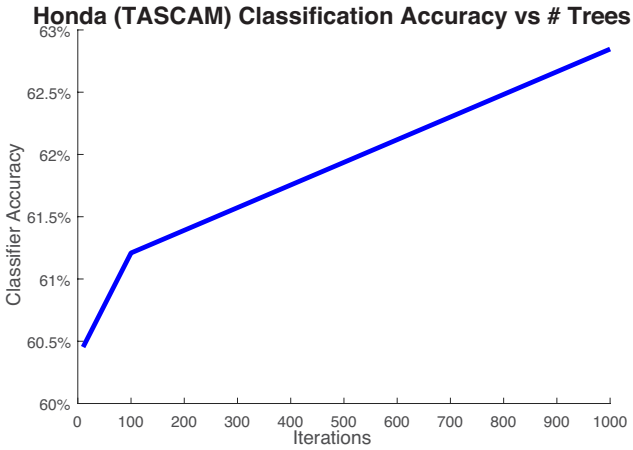


Figure 15: This plot shows how the algorithm's insample classification accuracy varying with number of trees used for bagging. We see a peak accuracy occurring at 1000 iterations and an upward trajectory, but note that returns are diminishing with increasing trees.

Table 1: This table shows the optimized ensemble classifier's parameters and performance for the three models tested.

| **Vehicle** | **Recorder** | **Segment Time** | **Bin Width** | **Input Trees** | **Maximum Splits** | **Insample Accuracy** | **Outsample Accuracy** |
|---|---|---|---|---|---|---|---|
| **Honda** | TASCAM | 1 s | 0.25 Hz | 1000 | 5 | 62.85% | 74.24% |
| **Honda** | iPhone | 1 s | 0.75 Hz | 1000 | 15 | 40.42% | 76.61% |
| **Mazda** | iPhone | 2 s | 3 Hz | 1000 | 10 | 39.381 | 79.66% |

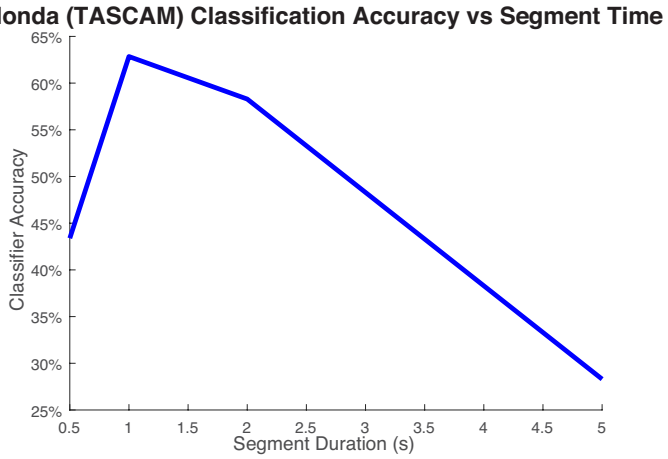


Figure 16: This plot shows how the algorithm's insample classification accuracy varies based on training and testing segment time. We see a maximum occurring at $t = 1.0$s. Using a relatively short time allows more segments to be used in training, cross-validation, and testing, thereby improving eventual robustness of the classifier.

Table 2: This table shows the features selected for use in classification as having non-zero values for MATLAB's PredictorImportance function.

| **Vehicle** | **Recorder** | **DFT Selected** | **DWT Selected** | **MFCC Selected** | **Features Selected** | **Features Total** | **Reduction** |
|---|---|---|---|---|---|---|---|
| **Honda** | TASCAM | 352 | 5 | 1496 | 1853 | 14452 | 87.18% |
| **Honda** | iPhone | 526 | 23 | 3572 | 4121 | 5575 | 26.08% |
| **Mazda** | iPhone | 120 | 20 | 2342 | 2482 | 3374 | 26.44% |

in differentiating states and that the features of interest are focused on narrow spectral regions. We also note that segment times tend to be short, providing each model with additional training samples. Finally, we see that each model uses the maximum-allowed $1,000$ trees.

In Table 2, we see the features and types selected as important by the classifier. We also highlight the percent of zero-importance features that were able to be eliminated.

Here, we see that each model uses a significant number of DFT and MFCC

Table 3: This table shows the percentage of each feature type used by the final classifier. Note that the total number of features changes based on DFT bin width and segment time.

| | **DFT Selected** | **DWT Selected** | **MFCC Selected** |
|---|---|---|---|
| **Honda (TASCAM)** | *18.3%* | *15.2%* | *12.0%* |
| **Honda (iPhone)** | *83.9%* | *69.7%* | *72.9%* |
| **Mazda (iPhone)** | *75.5%* | *60.6%* | *73.6%* |

Table 4: This table shows the Honda's performance with the stationary microphone, and demonstrates a strong diagonal (correct) component. Note that dirty and filthy filters are never mistaken for clean in this example.

| **Honda (TASCAM)** | *Clean* | *Dirty* | *Filthy* |
|---|---|---|---|
| *Clean* | **97** | 14 | 11 |
| *Dirty* | 0 | **52** | 24 |
| *Filthy* | 0 | 19 | **47** |

elements, with fewer DWT elements. Interestingly, the stationary microphone was able to eliminate a higher percentage of features. This suggests that either the TASCAM microphone has reduced system noise or that the moving models are classifying based on features other than those related to the air filter's performance.

We examine these feature types' inclusion in Table 3.

Note that the TASCAM audio input allows the algorithm to select fewer features across the board. FFT features a higher inclusion rate relative to the MFCC and DWT elements for this model and for the Honda's iPhone model.

In identifying mechanical system faults, false positives and negatives are important to understand. We consider the outsample confusion matrix to determine the number of false positives (reports dirty or filthy when actually clean) and false negatives (reports clean when actually filthy or dirty). These results are shown in Tables 4-6.

The overall performance and false positive rate (clean filter reported as non-

Table 5: This table shows the Honda's performance with the moving iPhone, and demonstrates a similarly-strong diagonal component. Here again we see a slight bias towards reporting a clean filter as dirty.

| **Honda (iPhone)** | *Clean* | *Dirty* | *Filthy* |
|---|---|---|---|
| *Clean* | **182** | 51 | 38 |
| *Dirty* | 5 | **97** | 12 |
| *Filthy* | 1 | 9 | **101** |

Table 6: This table shows the Mazda's performance with the moving iPhone, and demonstrates another strong diagonal. Once more, we see a bias towards reporting clean filters as being dirty.

| **Mazda (iPhone)** | *Clean* | *Dirty* | *Filthy* |
|---|---|---|---|
| *Clean* | **66** | 18 | 1 |
| *Dirty* | 9 | **49** | 5 |
| *Filthy* | 3 | 12 | **73** |

Table 7: This table summarizes each model's performance and false positive/false negative rate.

| Vehicle | Recorder | Outsample Accuracy | False Positives | False Negatives |
|---|---|---|---|---|
| **Honda** | TASCAM | 74.24% | 9.47% | 0.00% |
| **Honda** | iPhone | 76.61% | 17.94% | 1.21% |
| **Mazda** | iPhone | 79.66% | 8.05% | 5.08% |

clean) and false negative rate (non-clean filter reported as clean) are shown in Table 7. We prefer a slight bias towards false positives as air filters are low-cost and easy replacement, while dirty filters have significant cost implications.

Finally, our results show us that switching from a stationary to a moving microphone has a minimal change in accuracy when classifying the vehicle – in fact, the accuracy improves about 2.4% when using the iPhone instead of the TASCAM recorder. We also note similar predictor performance across vehicles, with a final difference in classification accuracy of approximately 3.1%.

## 7. Conclusions

We demonstrated 80% accuracy in three-state air filter particulate loading detection using MFCC, DFT and wavelet features and bagged decision trees, proving the viability of batch processed smartphone audio for filter classification. Multi-state classification is a step towards condition monitoring, while the demonstrated classifier's sensitivity suggests early response is possible. A mobile application using this approach may ultimately improve vehicle performance and efficiency.

The results are promising, but much remains to improve the algorithm. We intend to conduct an experiment to determine the optimal filter replacement and will continue development using an off-vehicle intake system instrumented with flow meters and controlled fans. This will enable us to observe how environmental conditions such as barometric pressure, or systemic changes such as microphone type and location will impact the classifier's accuracy.

Related work will examine the algorithm's applicability to forced induction engines and consider online implementations. We will further consider how to optimally generate "fingerprints" on mobile devices while conserving computation, bandwidth, and storage. This approach will allow computationally-intensive classification to run in the Cloud while delivering accurate results to an end user's constrained device. Ultimately, such a platform may be used to identify other maintenance needs from cabin air filters to faults in the exhaust system and beyond.

## References


[1] United States Department of Transportation, Federal Highway Administration, December 2015 traffic volume trends (2016).
URL https://www.fhwa.dot.gov/policyinformation/travel_monitoring/15dectvt/

[2] IHS Inc, Aging vehicle fleet continues to create new opportunity for automotive aftermarket, ihs says (2016).
URL http://press.ihs.com/press-release/automotive/aging-vehicle-fleet-continues-create-new-opportunity-automotive-aftermarket

[3] United States Department of Transportation - Bureau of Transportation Statistics, National transportation statistics: Table 1-11: Number of u.s. aircraft, vehicles, vessels, and other conveyances (2016).
URL http://www.rita.dot.gov/bts/sites/rita.dot.gov.bts/files/publications/national_transportation_statistics/html/table_01_11.html

[4] U.S. Energy Information Administration, Annual energy outlook 2016: Figure mt-25 (2016).
URL http://www.eia.gov/forecasts/aeo/pdf/0383(2016).pdf

[5] T. Jaroszczyk, J. Wake, M. J. Connor, Factors affecting the performance of engine air filters, Journal of Engineering for Gas Turbines and Power 115 (4) (1993) 693–699.

[6] K. Norman, S. Huff, B. West, Effect of intake air filter condition on vehicle fuel economy, ORNL/TM-2009/021, February.

[7] M. Toma, Investigating maintenance procedures for engine air filters, in: Proceedings of the European Automotive Congress EAEC-ESFA 2015, Springer, 2016, pp. 375–384.

[8] J. Thomas, B. West, S. Huff, K. Norman, Effect of intake air filter condition on light-duty gasoline vehicles, Tech. rep., SAE Technical Paper (2012).

[9] M. Toma, C. Bobalca, Research on drivers' perception on the maintenance of air filters for internal combustion engines, Procedia Technology 22 (2016) 961–968.

[10] J. Engelbrecht, M. J. Booysen, G.-J. . J. van Rooyen, F. J. Bruwer, Survey of smartphone-based sensing in vehicles for intelligent transportation system applications, IET Intelligent Transport Systems 9 (10) (2015) 924–935.

[11] Q. Han, D. Cho, Characterizing the technological evolution of smartphones: insights from performance benchmarks, in: Proceedings of the 18th Annual International Conference on Electronic Commerce: e-Commerce in Smart connected World, ACM, 2016, p. 32.

[12] J. E. Siegel, S. Kumar, I. Ehrenberg, S. Sarma, Engine misfire detection with pervasive mobile audio, in: Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases 2016, Springer International Publishing, Cham, 2016, pp. 226–241. doi:10.1007/978-3-319-46131-1_26.

[13] P. Davies, K. R. Holland, Ic engine intake and exhaust noise assessment, Journal of Sound and Vibration 223 (3) (1999) 425–444.

[14] S. N. Dandare, Multiple fault detection in typical automobile engines: A soft computing approach, WSEAS Transactions on Signal Processing 9 (3) (2013) 158–166.

[15] A. Sujono, Utilization of microphone sensors and an active filter for the detection and identification of detonation (knock) in a petrol engine, Modern Applied Science 8 (6) (2014) p112.

[16] P. Kabiri, A. Makinejad, Using PCA in acoustic emission condition monitoring to detect faults in an automobile engine, in: 29th European Conference on Acoustic Emission Testing (EWGAE2010), 2011, pp. 8–10.

[17] P. Kabiri, H. Ghaderi, Automobile independent fault detection based on acoustic emission using wavelet, in: Singapore International NDT Con-

ference & Exposition 2011, Singapore International NDT Conference & Exposition, Singapore, 2011.

[18] J.-D. . D. Wu, C.-H. . H. Liu, Investigation of engine fault diagnosis using discrete wavelet transform and neural network, Expert Systems with Applications 35 (3) (2008) 1200–1213. doi:10.1016/j.eswa.2007.08.021.
URL https://doi.org/10.1016/j.eswa.2007.08.021

[19] S. K. Yadav, K. Tyagi, B. Shah, P. K. Kalra, Audio signature-based condition monitoring of internal combustion engine using fft and correlation approach, IEEE Transactions on Instrumentation and Measurement 60 (4) (2011) 1217–1226. doi:10.1109/TIM.2010.2082750.

[20] A. K. Kemalkar, V. K. Bairagi, Engine fault diagnosis using sound analysis, in: 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 2016, pp. 943–946. doi:10.1109/ICACDOT.2016.7877726.

[21] J. E. Siegel, R. Bhattacharyya, S. Sarma, A. Deshpande, Smartphone-based wheel imbalance detection, in: ASME 2015 Dynamic Systems and Control Conference, American Society of Mechanical Engineers, 2015, pp. V002T19A002–V002T19A002.

[22] J. E. Siegel, R. Bhattacharyya, A. Desphande, S. E. Sarma, Smartphone-based vehicular tire pressure and condition monitoring, in: Proceedings of SAI Intelligent Systems 2016, 2016.

[23] Y. Jin, Z.-y. . Y. Hao, X. Zheng, Comparison of different techniques for time-frequency analysis of internal combustion engine vibration signals, Journal of Zhejiang University-SCIENCE A 12 (7) (2011) 519–531. doi:10.1631/jzus.A1000384.
URL https://link.springer.com/article/10.1631%2Fjzus.A1000384

[24] J. Siegel, R. Bhattacharyya, A. Deshpande, S. Sarma, Vehicular engine oil service life characterization using on-board diagnostic (OBD) sensor

data, in: IEEE SENSORS 2014 Proceedings, IEEE, 2014, pp. 1722–1725. doi:10.1109/ICSENS.2014.6985355.

[25] C. Zhang, Y. Ma, Ensemble machine learning, Springer, 2012.

[26] J. Zhu, H. Zou, S. Rosset, T. Hastie, Multi-class adaboost, Statistics and its Interface 2 (3) (2009) 349–360.

[27] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32. doi:10.1023/A:1010933404324.
URL http://dx.doi.org/10.1023/A:1010933404324

[28] D. Che, Q. Liu, K. Rasheed, X. Tao, Decision tree and ensemble learning algorithms with their applications in bioinformatics, in: Software Tools and Algorithms for Biological Systems, Springer, 2011, pp. 191–199.

[29] D. Zhang, X. Zhou, S. C. Leung, J. Zheng, Vertical bagging decision trees model for credit scoring, Expert Systems with Applications 37 (12) (2010) 7838–7843.

[30] D.-I. Curiac, C. Volosencu, Ensemble based sensing anomaly detection in wireless sensor networks, Expert Systems with Applications 39 (10) (2012) 9087–9096.

[31] A. Kelarev, A. Stranieri, J. Yearwood, H. F. Jelinek, Empirical study of decision trees and ensemble classifiers for monitoring of diabetes patients in pervasive healthcare., in: 2012 15th International Conference on Network-Based Information Systems, 2012.

[32] P. Bühlmann, Bagging, boosting and ensemble methods, in: Handbook of Computational Statistics, Springer, 2012, pp. 985–1022.