Vibrational Event Detection and Classification in Goodwin Hall

By Henry Grozier, Jordan Hulbert, Tim Koskulitz, Matthew Ogden, and Jackson Ray

With Dr. Mark Embree of VT Mathematics and Dr. Rodrigo Sarlo of VT Civil Engineering

May 3rd, 2023

Executive Summary

Good-win Vibrations has been tasked by Dr. Rodrigo Sarlo and Dr. Mark Embree with utilizing time-series accelerometer data collected in Goodwin Hall to detect and classify "events". Specifically, our team was focused on detecting footsteps in the building. The motivation behind footstep detection is the development of "smart buildings" across campus, or buildings that incorporate similar technologies to those in Goodwin Hall. The data collected within these buildings can be utilized to automate the internal building functions such as heating and cooling, controlling when and which lights are on and ultimately increasing the energy-efficiency of the building. We used an internal data acquisition device [DAQ], located inside of Goodwin Hall, to collect the datasets. To "clean" these datasets, we utilized a high-pass filter to remove any noisy trends and make our datasets easier to work with. We utilized the *highpass()* function in MATLAB, which is part of the *signal processing toolbox*. After cleaning, our group created an event detection model that utilized kurtosis as its underlying analytical method. Using kurtosis, our model recorded time windows from the datasets which represented "events". These events were then classified through principal component analysis [PCA]. This classification model discerned between footsteps and other events in the building. Although our PCA model is limited to only accurately identifying footsteps, our work has introduced new mathematical methods for our sponsors to consider in pursuit of completing a more robust classification model.

1 Problem Statement

Our team, Good-win Vibrations, analyzed footstep vibrations in Goodwin Hall in order to monitor pedestrian traffic throughout a given time frame. We measured vibrations from accelerometers, which are devices that collect vibrations from a change in motion. Identifying when an event took place, such as footstep vibrations, was our main objective, followed by classification of the individual footsteps in Good-win Hall. With the help of Dr. Mark Embree and Dr. Rodrigo Sarlo, our results are beneficial and can assist Virginia Tech into fully automating its internal functions. These functions, like Heating, Ventilation and Air Conditioning [H.V.A.C.] systems turning on/off and doors closing, are important factors to consider when monitoring the pedestrian traffic patterns on a given day. This would help cut down on operating costs and improve the overall internal health of Goodwin Hall.

2 Ethical Considerations

Some ethical problems and or issues that could arise with our project is the misuse of our data and data limitations within the population, such as sampling bias from the given data. When considering the ethical issue of misusing our data, it is possible for someone to gain access to our data without permission, and they could use that data in order to cause harm or violence among an individual or a group of people inside Goodwin Hall. Since our goal is to classify footsteps and monitor pedestrian traffic, our results will include a heatmap displaying the most populated locations inside the academic building at a given time. If an individual knows the most populated areas of the building, they may use that information to carry out violent attacks or track groups of people throughout a given day. In order to fix and address the ethical problem of tracking a group of people in Goodwin Hall, we made sure our data was only accessible to our team, our sponsors, and trusted Virginia Tech students and staff. The second ethical issue that can occur pertains to data limitations in the population. For example, our project uses accelerometers to detect certain vibrations that can vary depending on the intensity and duration of the physical activity taking place, and this can be a concern for individuals who are disabled, such as those in a wheelchair. The vibration of the person in the wheelchair may be misclassified since the wheels on the ground produce different vibrations than footsteps. In order to resolve this issue, our team has distinguished footsteps from non-footsteps in the classification process and made sure to account for non-footstep vibrations produced by different members of our team.

3 Literature Review

Our team began researching about ways in which we can determine events and apply various techniques to the accelerometer data. Some sources found that assist us prior to working on the project itself are described below.

One big goal of our project is to be able to use patterns of foot traffic in Goodwin Hall in order to create a more energy efficient schedule for it's internal functions. Cory D. Kidd, Et al.,[1] talks about the effectiveness of using the data acquired from well-equipped smart buildings in order to solve real world problems, such as cutting down on costs or preserving the overall internal health of a building. From this paper, we have learned multiple methods in which the data acquired from these smart buildings can be applied to solving similar problems to the problems we are trying to solve.

Accompanying the event detection problem is event localization. With 236 accelerometers welded to the frame of the building, many of these devices are going to be recording the same signal. Vu, et al. [2] discusses localized event detection through, visualized using clustering methods such as k-nearest-neighbors. This method ties into our approach for visualization as we have discussed the possibility of clustering localized events to their nearest respective accelerometer. Clustering data based on accelerometer readings shall likely come after some more in depth analyzations such as Independent Component Analysis, but this paper gives us a good insight on how we could use clustering methods to create effective visualizations to help convey our later findings.

There are many different techniques we can use in order to clean the signal data. The accelerometers may experience a drift or some other unwanted frequency. Imperfections in the instrumentation, as well as weather, almost always cause some noise within the data. Safak and C. akti [3] explain how they used types of bandpass-filtering in order to minimize the effects of vibrations caused by structural components of the building. We consider using lowpass-filters, highpass-filters, and bandpass-filters built into Matlab in order to dampen out some of these unwanted effects that cause noise in the data. This way, we are working only with data that is created by procedural events, such as a person walking or a door closing.

4 Project Criteria

The project was split up into clear components, each of which having a specific role in the process of this project. These components include criteria, or a set of pieces of the component necessary for success. These criteria are as follows:

- Data Acquisition
  - Data Validation- Our data had to be large enough and diverse enough to give an accurate analysis of what is occurring in the building.
  - Data Visualization- We modeled the data based on where vibrational events were occurring, quantified by the strength of the signals.
  - Data Cleaning- We made sure the the data being used was cleaned and that things such as trends, drift, and noise were accounted for.
  - Format of Data Set- The format of the data set had to be of a specific size and frequency to be accurately analyzed.
- Event Identification and Analysis
  - Event Detection- We pulled the vibrational events out of the time series data, basing accuracy on the comparison of the results from the model and the real-life occurrences during in-person testing.
  - Event Classification Model- The detected events were classified by their cause, again using the in-person testing script to signify the accuracy of the model.
  - Accuracy- Once all of the event detection and classification had been completed, the data was parsed through to see if each accelerometer was able to accurately predict the cause of each event.

5 Selected Solutions

5.1 Kurtosis & Root Mean Squared Error

For event identification and analysis, our team worked with both kurtosis and root-mean squared approaches for detecting events in Goodwin Hall. Kurtosis is a measure of the tailed-ness of a distribution, and in our case, we analyzed the kurtosis values from varying signals to see where the most "random" events are occurring and observed when the kurtosis values were non-Gaussian. For kurtosis, we set a threshold for when the kurtosis value was greater than 1, and the equation we used is defined below.

$$k(x) = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^4}{(\sum i=1^n(x_i-\bar{x})^2)^2} - 3$$

n = number of samples

x_i = acceleration at time i

$\bar{x}$ = mean acceleration

In the equation, the numerator represents the kurtosis, or fourth moment, and the denominator is the equation for the variance squared. This method detects outliers in the data and we were able to clearly detect when there were differences from the Gaussian distribution when the kurtosis values were greater than 1.

The root-mean squared approach, on the other hand, calculates the average value of a signal over a given time, and its equation is defined below.

$$RMS = \sqrt{\tfrac{1}{n}\sum_{i=k-n}^{k} y_i^2}$$

n = number of samples

k = kth RMS value

y = filtered acceleration

With the root-mean-squared approach, our team was able to detect events when observations were above a given threshold. However, kurtosis proved to be the best solution in our case for determining events in our data.

5.2 Principal Component Analysis

For the classification portion of the project, our team used principal component analysis and utilized k-means clustering for differentiating between footsteps/non-footsteps in the data. From our data, we used our event detection methods to form a matrix of events. This will be discussed in more detail later on, but essentially each row of this matrix contains information about events that we're detected using the kurtosis event detection method.

Our event matrix is empirical data, so we decided the best method for classifying our sample data is to perform an approximate principal component analysis. Our sponsor Dr. Mark Embree wrote the textbook that contains the method of approximate principal component analysis that we followed for this project.

Suppose our event matrix had columns containing various features regarding each event. These columns are random variables, $X_1, \ldots, X_n$ with m samples each:

$$x_{j,k}\text{k} = 1, \ldots m,$$

The expected value has the unbiased estimate

$$\mu_j = \frac{1}{m}\sum_{k=1}^{m} x_{j,k}$$

We can also approximate the sample covariance matrix with the equation

$$\mathbf{x}_j = \begin{bmatrix} x_{j,1} \\ x_{j,2} \\ \vdots \\ x_{j,m} \end{bmatrix}, j = 1, \ldots, n$$

The kth sample principal component is given by $\mathbf{v}_k^T \mathcal{X}$. In our project, our principal components' scale changes each column. This is because we are taking physical measurements and the units for each column are different. We will need to normalize our eigenvalues before we make any decisions about the features or principal components in order to get rid of this bias in our data. Each entry of the eigenvector $\mathbf{v}_k =$

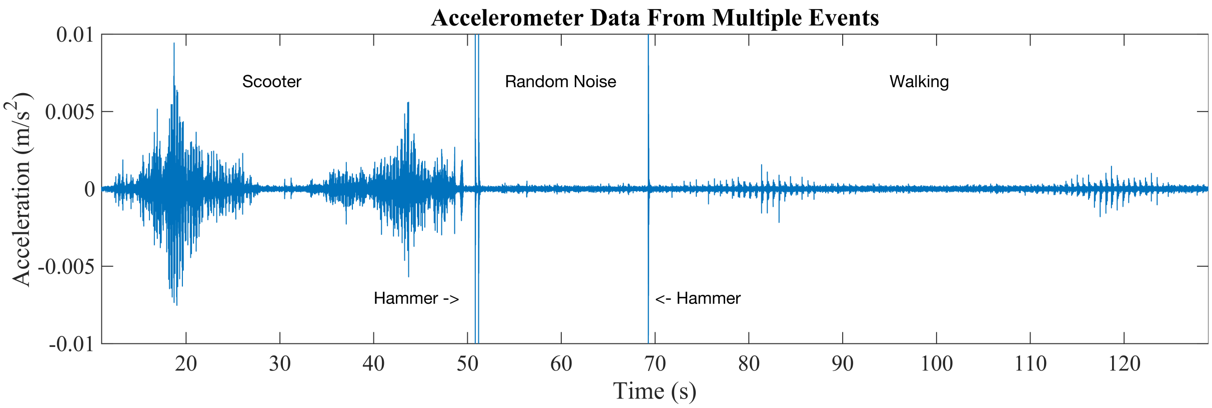


Figure 1: First Two Minutes of Data Recorded on April 1st, 2023

$k = 1, \ldots, m$ corresponds to a feature vector X_k in X. That scalar tells us how much effect the corresponding feature at index k has on the kth principal component. So the entry in $\mathbf{v}_k$ with the largest value, has the most effect value of the kth principal component.

The importance of the various principal components can be seen via the eigenvalues of $\boldsymbol{S}$, given by $\lambda_k = \sigma_k^2/(m-1)$. If our eigenvalues decrease rapidly, we can assume that most of our data can be explained by the first few principal components. We will select the first k principal components to use for our classification model based on how much of the total variation is contained in the first k components. Choosing k relies on the assistance of a scree plot which we use for our data later in this paper.

6 Results

6.1 Team-Developed Data

The majority of the data that our team has worked with at the beginning of the semester had been data that was previously collected from Goodwin Hall by Dr. Sarlo and his colleagues. However, in order to properly understand what the vibrational pattern of certain events looks like, we decided to obtain some of our own data from the heavily-instrumented hallway of the building, or the hallway with the largest quantity of accelerometers. We tested multiple types of events occurring in the hallway including: walking with both soft and hard sole shoes, hitting a hammer on the ground, riding a scooter, running, and walking barefoot. We expect each of these events to be able to be differentiated from each other based on the data collected from the accelerometers. So, we put this data we collected through a high pass filter, in order to account for trends, drift, and outside noise in the time series data and plotted a section of it below.

Looking at **Figure 1** does not make sense to the untrained eye until the script is explained. For reference, a larger deviation from zero on the y-axis indicates a larger amount of vibrations at the location of the sensor at that time. In this time window, the order of events are as follows:

1. Henry riding a scooter up the hallway and then back down the hallway (seconds 10-52)
2. Two hammer strikes indicating the end of a section of events (second 52 and 53)
3. One hammer strike indicating the beginning of a section of events (second 68)
4. Tim wearing hard-soled shoes walking up the hallway and then back down the hallway (seconds 68-130)

Figure 1 makes it easy to visualize what is happening in real-time.

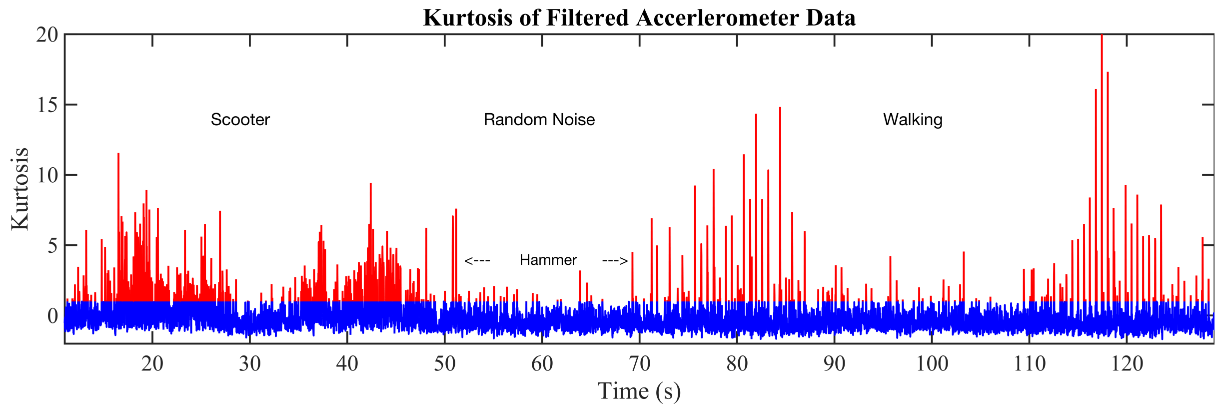


Figure 2: Kurtosis values during multiple events

We also recorded eight more minutes of data that we use for event detection and classification. The other events that we may suspect to see throughout our entire data set but are not within the first two minutes of testing include Jordan using crutches while walking down the hallway, all five group members walking down the hallway at the exact same time, as well as group members running and skipping down the hallway at various speeds.

Next, we will apply the kurtosis methods that we previously used on the data that was provided to us, to see if it would both accurately identify when events are occurring and if it would show differences between what is a footstep and what may be another event. The data was collected at a different rate than the previously collected data, so the code was modified for a data collection rate of 1024 samples collected per second, a kurtosis window size of 32 samples, and a window overlap size of 16 samples. The kurtosis plot of the same snippet of filtered accelerometer data can be see in **Figure 2**

Each red value indicated a kurtosis value larger than the threshold that signifies an event, while the blue values remained below this threshold. In our case, the threshold is 1, as this signifies a curve steeper than the traditional bell curve in the window of this kurtosis graph. Each red spike in the data represents an event i.e. a footstep, a hammer strike, et cetera. The kurtosis data showed some valuable insight on the effect of certain events on the accelerometer.

6.2 Event Detection

We decided to perform our event detection using the kurtosis equation explained in 5.1. With the use of this equation, we established that we could determine when an event occurs based on the kurtosis values, because whenever we have a kurtosis value above 1, this tells us that something non-gaussian is occurring. Using this logic, we established a model which detects events by first grabbing sections of data which we call "windows", each 32 samples long, and calculating the kurtosis of that window. The value calculated from the window gives us a single value that is representative of the entire window of data. Once we have recorded the kurtosis values over our entire dataset, we are ready to look for events occurring based on our operational definition of what an event is:

- There must be at least 8 windows of kurtosis values less than 1 before the event
- There must be at least 3 windows of kurtosis values greater than 1 during the event
- There must be at least 8 windows of kurtosis values less than 1 after the event

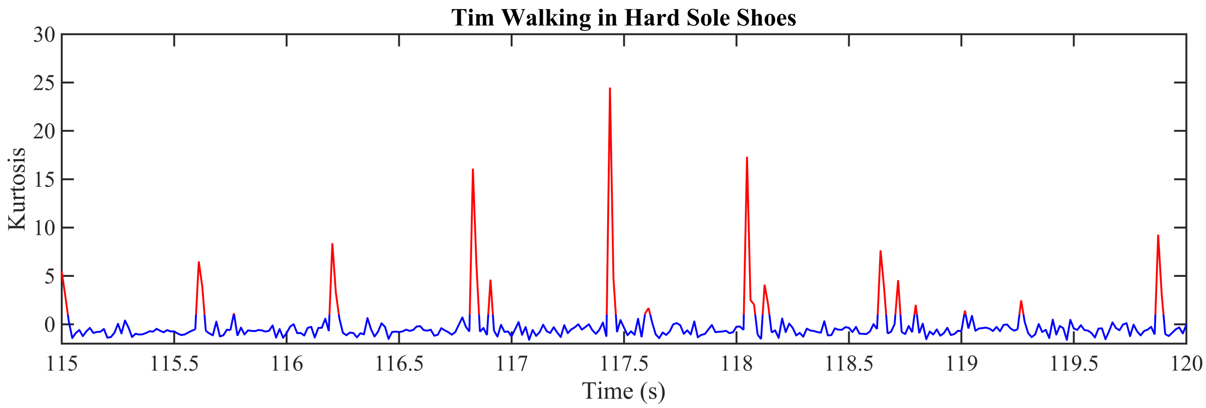


Figure 3: Kurtosis values from hard sole shoes test

By following this definition, we can observe the **Figure 3** and decide exactly what we are going to determine is an event versus just random noise:

Observe how there are portions of the plot where you can see a large spike followed by a smaller spike in kurtosis. Since this entire section is Tim walking, our intuition would tell us that the large spike must be the heel hitting the ground followed by the toe hitting with a smaller spike. This is important to note because we intend on classifying each event as just one footstep instead of two separate steps. This is why our operational definition is so important in helping us decipher when events begin and end.

Now that we have determined where our events are occurring, we decided to create matrices to store all of the information related to each event, including one matrix with all of the kurtosis values during each event, one with the exact times each event occurred, and one with the original frequency values for each event. These matrices prove to be useful as we build our classification model and an animated heatmap that details the general location of events happening within the building.

Figure 4: Heat Map which glow orange when an event is detected

In **Figure 4**, our animated heatmap displays accelerometers in a heavily instrumented hallway on the fourth floor of Goodwin. A base color of black signifies that there was no event detected and a base color of orange signifies that there was an event detected. Using our detection method, original frequency matrix, and the time windows of the events are recorded, the accelerometers are changed to reflect when an event happens. As stated the accelerometer color reflects when an event occurred and its size reflects the frequency of the signal that was recorded. Instead of scaling the size at every time stamp, the size only changed during an event.

6.3 Event Classification

We performed classification on the events that we collected using principal component analysis. We had various ideas as to which features to use when classifying events. Our group came up with 6 features of an event that we assumed would have a strong effect on our classification model. The features of each event that are included in our matrix X are:

- X_1: The minimum amplitude is subtracted from the maximum amplitude of an event.
- X_2: The frequency of an event when acceleration is at its maximum, is obtained by the Fast Fourier Transform function (FFT). This essentially takes our signal and breaks it down into sin and cosine waves with different frequencies. The frequencies that make up our signal have the highest values.

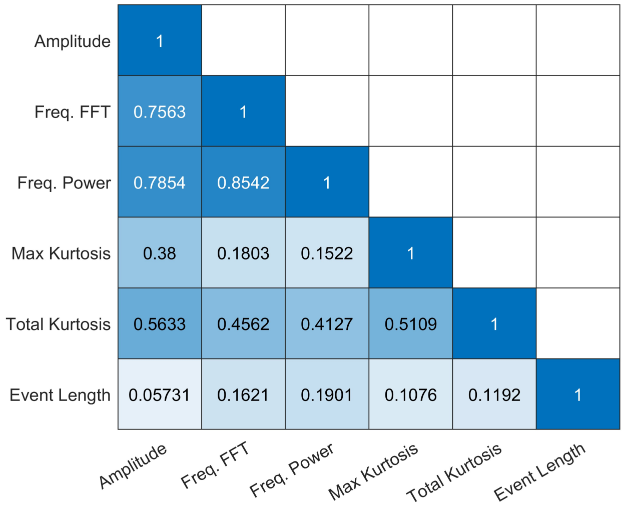


Figure 5: Correlation Matrix with 6 Features

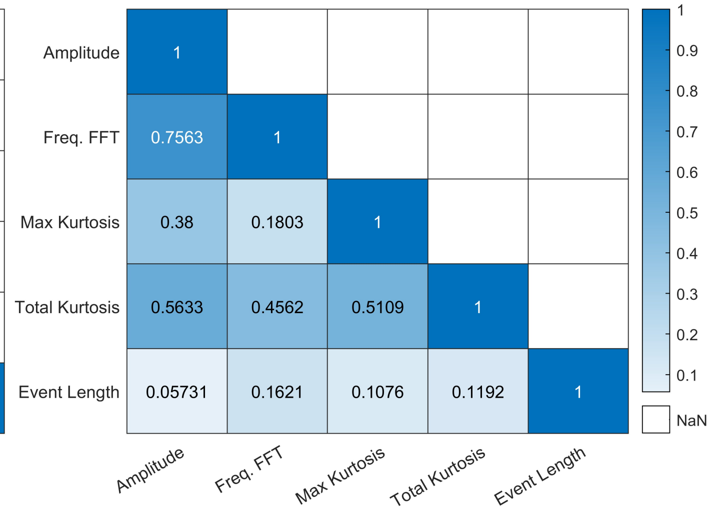


Figure 6: Correlation Matrix with 5 Features

- X_3: The frequency of an event when acceleration is at its maximum, is obtained by the Power Spectral Density function (Pwelch).
- X_4: The maximum kurtosis of one of the windows that make up an event
- X_5: The sum of the kurtosis across all the windows that make up a single event
- X_6: The number of windows that make up an event.

In **Figure 5**, we show a correlation matrix between the six above features. We noticed that there is a high correlation between the frequency from the Fast Fourier Transform and the frequency from the Power Spectrum. These two features are returning the frequency that makes up the signal recorded by our accelerometer. Both these functions return similar frequencies for each event. This caused over-fitting of our classification model, so we decided to remove one of the two correlated features. We decided to keep the Fast Fourier Transform and remove the Power Spectrum. We used the Fast Fourier Transform because the Fast Fourier Transform tends to give us better results for events that are shorter in length. The Correlation matrix for the five features can be seen in **Figure 6**. We will be using these five features from here on out.

We reconstructed our X matrix without feature X_3 and constructed a mean-centered data matrix $\mathcal{X}$. With this mean-centered matrix, we created the symmetric covariance matrix. However, each column in $\mathcal{X}$ is scaled differently, so we must go back and normalize the columns in $\mathcal{X}$ for more precise results. In order to do this, we divided each column vector by the norm of the corresponding vector. We then constructed a covariance matrix $\boldsymbol{S}$ and computed the eigenvalues of $\boldsymbol{S}$. Each λ_k assesses the importance of the kth principal component.

We computed the eigenvalues of the sample covariance matrix. Note the matrix of corresponding eigenvectors (V) and vector of sorted eigenvalues (λ) are reported below as well. Notice that there are only 5 eigenvalues because we removed the feature X_3 from the matrix we used for PCA.

$$\lambda = \begin{bmatrix} 0.1310 \\ 0.0497 \\ 0.0217 \\ 0.0142 \\ 0.0105 \end{bmatrix}, V = \begin{bmatrix} -0.5180 & -0.0062 & 0.2574 & 0.0546 & 0.8139 \\ -0.7565 & 0.3896 & -0.1514 & -0.2897 & -0.4112 \\ -0.2187 & -0.8598 & 0.0973 & -0.4260 & -0.1480 \\ -0.3269 & -0.2797 & 0.0763 & 0.8510 & -0.2913 \\ -0.0687 & -0.1750 & -0.9463 & 0.0859 & 0.2484 \end{bmatrix} \tag{1}$$

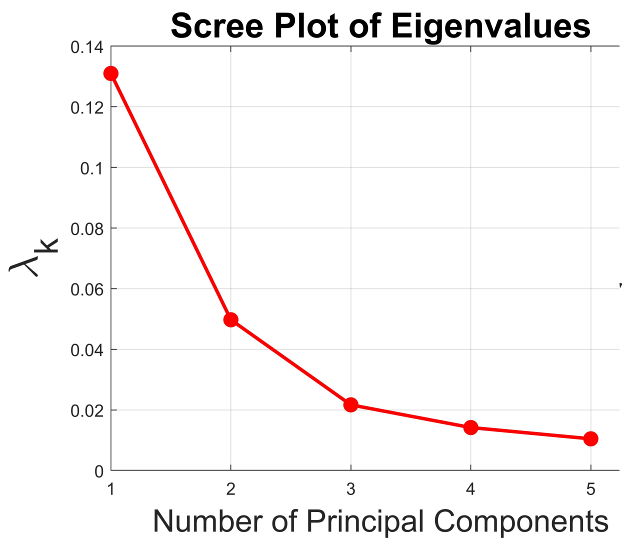


Figure 7: Scree Plot of Eigenvalues

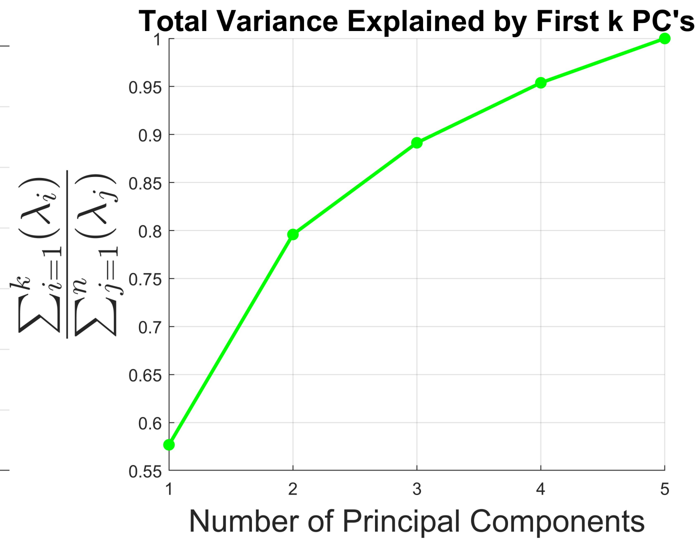


Figure 8: Correlation Matrix with 5 Features

The first and largest eigenvalue tells us the importance of the first principal component, the second eigenvalue tells us the importance of the second principal component, and so on. We plotted these ordered eigenvalues on a scree plot in **Figure 7**. Scree plots are common visualizations used by data scientists to decide how many principal components are worthy of consideration, and after analyzing our scree plot, we picked the eigenvalue located at the elbow of the plot in **Figure 7**. The number of principal components we chose to consider for this project is $k = 3$. **Figure 8** tells us how much variation is explained by the first k principal components. Since the first 3 principal components explain 90 percent of the variation, and $k = 3$ is also the elbow point of the scree plot, we chose 3 principal components to perform k-means clustering on. Notice if k were to equal 2, then 80 percent of the total variation would be explained by the first two principal components, making $k = 2$ also a valid number of principal components to choose from.

The kth principal component is affected by the kth eigenvector. In other words, the element in $\mathbf{v}_k$ with the greatest absolute value is the feature that affects the kth principal component the most. For example, the principal component that holds the most total variance of the sample data is the first one, since it has the largest associated eigenvalue λ_1. The second entry of $\mathbf{v}_1$ is $|-0.75| = 0.75$, which is the largest entry in $\mathbf{v}_1$. This means that the feature X_2 (the frequency of the maximum acceleration returned from the Fast Fourier Transform) has the greatest effect on our first principal component. We can repeat this process for the second most important feature of a principal component, or repeat for the principal components up to n.

We plotted the first three principal components in three dimensions, but it is difficult to pick out certain clusters on our own. So we decided to use k-means clustering in order to figure clusters in the data. The distance from each cluster centroid to each point is calculated. The points closest to the respective centroid are labeled based on the name of the cluster centroid. The first three principal components can be seen with and without clusters in **Figures 9** and **10**. In early attempts, we clustered our data into 2 clusters but noticed that there were more than two separate events within our data. We chose 4 clusters because we noticed that there seems to be a large cluster of events around (0,0,0) and the other three clusters appear in a line shooting out in the direction of one of the principal components.

The clusters created based on the first 3 principal components can be plotted in two dimensions to increase readability. Plotting the principal components of the events in two dimensions also allows us to implement the MATLAB-created function *ginput()*. The *ginput()* function allows us to click a point on the two-dimensional principal components plot. This function captures the first and second principal component values associated

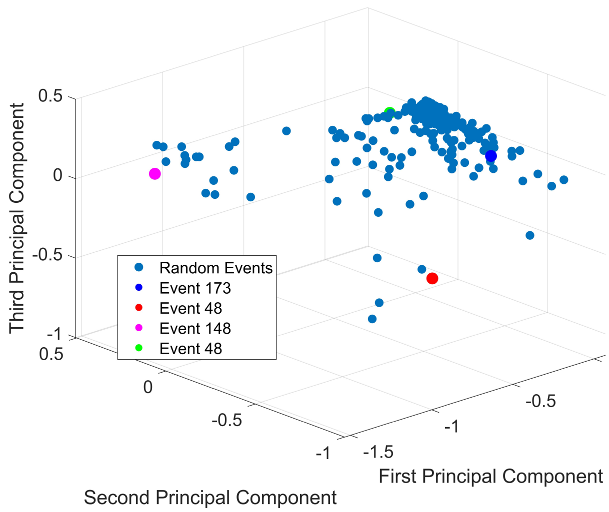


Figure 9: First 3 Principal Components Plotted with no Clustering

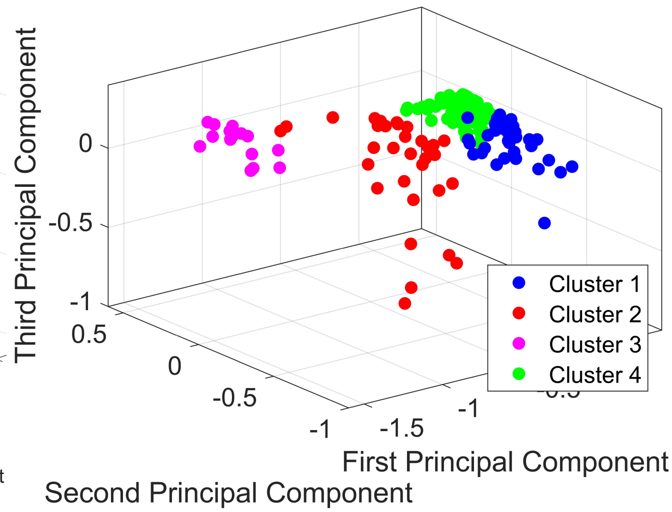


Figure 10: First 3 Principal Components Plotted with Colored Clustering

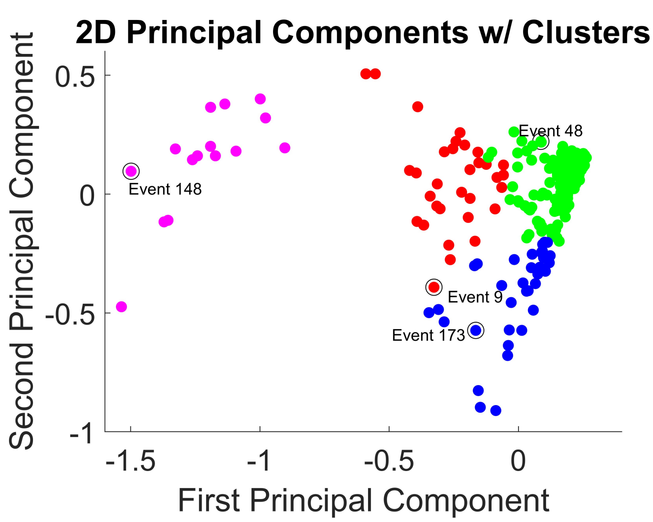
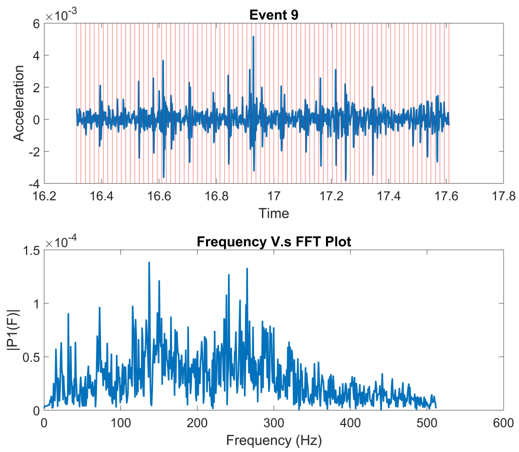
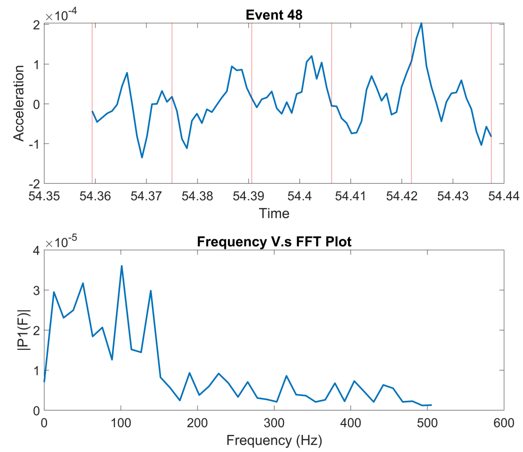


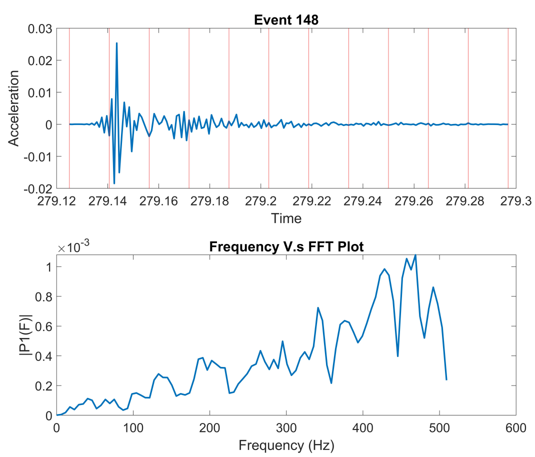
Figure 11: Principal Components Plotted in Two Dimensions



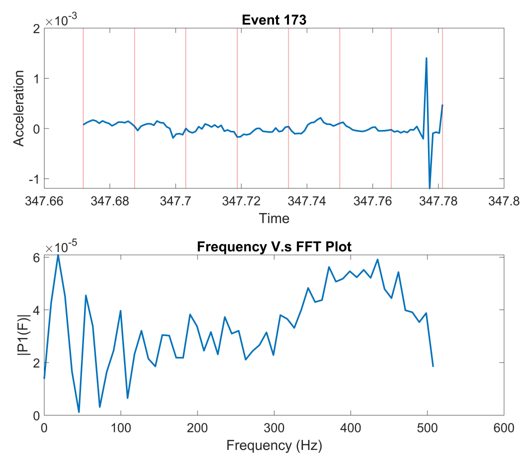
(a) Cluster 1: Unnatural event: Scooter



(b) Cluster 2: Shorter Scooter event



(c) Cluster 3: Hammer Strike



(d) Cluster 4: Suspected Footsteps

Figure 12: Arbitrary Events Captured with *ginput* from Each of the Four Clusters Captured and the Associated Event's Fast Fourier Transform.

with the event we clicked on and plots the signal of the event, as well as the frequency v. acceleration graph returned from the Fast Fourier Transform function. We chose four arbitrary points we click on in **Figure 11** to showcase our cluster features. These events are indexed at 9, 48, 148, and 173. We analyzed each event and go through our group's reasoning as to why each event belongs with each cluster based on the features we chose in our mean-centered data matrix. All four of these events can be seen as the colored points in **Figure 9**.

- Event 9 can be found in **Figure 12a** and appears to be a long event. The red lines in **Figure 12a** represent the time intervals at which our sliding window begins. The windows combined together create what we define as an event. Each window of red lines is the same length in each graph. Looking at the 3-dimensional principal component plot **Figure 9**, we can see that event 9 is located more in the 3rd principal component direction. Using our matrix of eigenvectors V (1), the amount of variation explained by the third principal component is influenced by the third eigenvector in V. The feature with the greatest magnitude in the column vector V_3 is the event length which is $|-0.9463| = .9463$. We can see that event 9 is farther in the direction of the third principal component than any other, which makes sense since there are 87 windows that create event 9.
- Event 48 can be found in **Figure 12b**. This event is short and only 6 windows long. The frequency is slightly larger than our footsteps. We are unsure what events in cluster 2 represent, but we suspect them to be scooter events that were captured. However, the kurtosis function was not able to collect the entirety of the event. We suspect this to be the scooter because the frequency spectrum plot returned by the FFT of event 48 looks similar to event 9's FFT. We perform model validation in a later section, however, future work may include a way to further verify our model's results.
- Event 148 can be found in **Figure 12c**. This event is short and has a maximum acceleration at a very high frequency from the FFT plot (468 Hz). Both of these features indicate to us that these are characteristics of the ground being hit with a hammer. The first principal component is most affected by the frequency of the event's signal, which can be seen in the 1st column of matrix V. The second feature (the FFT) has the most effect on the principal component that explains the most variation in the data.
- Event 173 can be found in **Figure 12d**. This event is again short and only eight windows long. Event 173 is located in the cluster with the most events. We expect most of the events to be footsteps. Since we spent the majority of the data collection period walking down the hallway, it makes sense that footstep events are the most frequent.

6.4 Model Validation

Our next step in the process would be to validate our classification model which utilizes k-fold cross-validation on our data. We split our feature matrix into training and testing data. During k-fold cross-validation, we essentially left k observations out of our training data set and had those observations be in our test data set. The reason we performed k-fold cross-validation is to confirm that the model would be able to make good predictions about new events not included in our data.

We used two methods, the naive method, and the approximation method. In both methods, we are essentially taking a test data point $\mathbf{x}^{(i)}$ at event i and we try to make a prediction $\hat{\mathbf{x}}^{(i)}$ about that event. We calculated the difference in the norm squared between the measured and the predicted test data points. This can be difficult to calculate, so instead, we used the eigenvectors from the matrix V to assist in our calculation. Our cross-validation methods follow the calculation of these two equations.

The Naive Norm Method:

$$||\mathbf{x}^{(i)} - \hat{\mathbf{x}}^{(i)}||^2 = \sum_{i=1}^{n} ||\mathbf{x}^{(i)} - \mathbf{V}^{(-i)}[V^{(i)}]^T \hat{\mathbf{x}}^{(i)}||^2 \qquad (2)$$

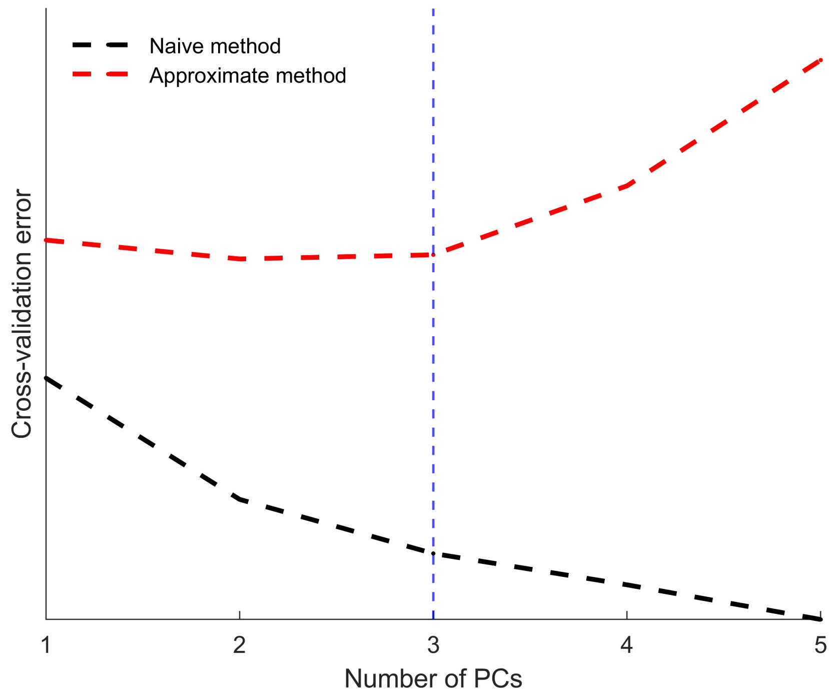


Figure 13: Cross Validation Error

The Approximation Norm Method:

$$||\mathbf{x}^{(i)} - \hat{\mathbf{x}}^{(i)}||^2 = \sum_{i=1}^{n} ||\mathbf{I} - \mathbf{V}\mathbf{V}^T + \text{diag}\{\mathbf{V}\mathbf{V}^T\}||^2 \tag{3}$$

We plotted the error in **Figure 13** and drew a vertical line at $x = 3$ because the principal components after the third do not have any effect on the norm-squared of the approximation method. Our naive function showed that all of the components are significant. The approximation method contradicts this because there is little change after the second and third components. This means our clustering method could theoretically be improved. In fact, using just two principal components gives us a smaller error than if we were to use three. So using two principal components makes our classification model just as accurate.

The error associated with the approximate method and naive method can be seen below for each of the principal components we added. The approximate method error can be viewed as the accuracy of our model. Our classification model will be useful in distinguishing time periods of heavy foot traffic in the building and random events that may occur throughout the building.

Method	PC1	+ PC2	+ PC3	+ PC4	+ PC5
Naive Error	23.7211	11.7982	6.4845	3.4060	0.0000
Approximate Error	37.2692	35.4183	35.8250	42.6022	54.9551

Due to the nature of our event detection model, an accelerometer may capture more than one event in the data. This may cause two events that occur simultaneously to be captured in one event. One of the two events that occur simultaneously will be classified correctly while the other will not be classified correctly.

7 Limitations on Deliverables

The primary limitation faced by our team was the minimal testing environment that the team used for data collection. Goodwin Hall is very large and there are many regions in which there are a limited number of accelerometers nearby, so for the sake of understanding our data we based the testing that we conducted in one hallway that is heavily instrumented with accelerometers. Although this was the correct decision for the scope of this project, it is important to note that the event detection and classification were performed on a smaller scope, and attempting to amplify the range of these techniques could potentially lead to some inaccurate results.

Another limitation is that we used the assumption that data from each accelerometer are relatively the same in scale. In reality, it is not likely that each accelerometer is fastened to the structure of the building slightly differently, causing different accelerometers to interpret the same event in slightly different ways. These differences are miniscule and it is unlikely that they cause significantly different results, but it is important to note this limitation prior to attempting to utilize this data.

8 Interpretation of Results for Client

The results we obtained are useful for figuring out how to identify significant vibrations throughout Goodwin Hall and classifying between different events in the building. With our animated heat map of the fourth floor in Goodwin Hall, the kurtosis event detection process, and the classification model using principal component analysis, Fast Fourier Transform, and k-means clustering, our client is able to understand the implications and procedure behind detecting and differentiating between footsteps/non-footsteps in the building. This would allow our client to observe the foot-traffic patterns on a given day granted that data is being collected throughout the day in Goodwin Hall and analyzed with their own data. These results have been delivered to our client in the form of a shared Google Drive. This drive includes related research articles that were fundamental in our development processes, MAT files of all of the processed time-series data separated by date and methodology, MATLAB scripts for both Event Detection and Classification, and a README file to explain how to run our MATLAB scripts and how to switch the MAT file.

9 Team Roles

Henry Grozier

Technical Contribution: Henry led the development of the event detection model which utilized the kurtosis function to create matrices and visualizations of events based on the operational definition.

Non-Technical Contribution: Henry presented midterm results, and assisted in both the Tech Memos the collection of data.

Tim Koskulitz

Technical Contribution: Tim collected data at Goodwin Hall and created a k-means clustering algorithm for event classification with Jordan in MATLAB.

Non-Technical Contribution: Tim presented for the Tools and Techniques Workshop, assisted in Tech Memos, and will present for part of the Final Presentation.

Jackson Ray

Technical Contribution: Jackson developed the event detection model with Henry which is used for the creation of the event matrix. He also helped to create the binary kurtosis matrix as well.

Non-Technical Contribution: Jackson presented during both the Elevator Pitch and the Midterm Presentations, as well as scripting the in-person data collection in Goodwin Hall.

Jordan Hulbert

Technical Contribution: Jordan led the development of our model's selected features during classification and created the kurtosis functions for event detection. Used various Data science techniques and visualizations to optimize the results from the classification of our principal components.

Non-Technical Contribution: Jordan introduced a variety of methods used during the Tools and Techniques and Midterm presentations. Assisted in the collection of data

Matthew Ogden

Technical Contribution: Matthew collected data at Goodwin Hall, refined operational definition of an Event used for the event detection model and refined kurtosis model that was utilized in final visualizations. He also used the final refined binary kurtosis matrix to create an animated heat map that displays estimated location of footsteps in Goodwin Hall.

Non-Technical Contribution: Matthew acted as our group's non-technical leader. He scheduled meetings, met with group members to check on progress and corresponded with sponsors.

10 Conclusions and Future Work

After collecting data and analyzing this data using various Event Detection techniques, our team has arrived at a few important conclusions.

Our first important conclusion is that not all Event Detection techniques are equally effective in this project's scope. Out of the proposed solutions, our team narrowed down what we considered to be the two most effective methods - kurtosis and root mean squared error [RMS]. We narrowed down these two methods by evaluating simplicity of implementation and flexibility of the method. RMS was the more simple method to implement of the two. However, the kurtosis method was far more flexible in terms of statistical properties. After creating an Event Detection model for each underlying method, we found that kurtosis was detecting events far more accurately than the RMS method.

Similarly, not all Event Identification techniques are equally effective in this project's scope. After researching multiple Event Identification methods, our group decided on pursuing two methods - principal component analysis [PCA] and fast fourier transforms [FFT]. Much of the research that our group conducted revealed that PCA and FFT were equally effective, our group chose to pursue PCA the furthest of the two methods. This is directly due to the fact that PCA is a very "malleable" algorithm. We could manually choose the number of components we wanted to use and evaluate exactly how it would effect our results.

In terms of future work, Dr. Rodrigo Sarlo was discussing with us about potentially having a future capstone team picking up from where our group ended. This future extension could involve a future capstone team creating a more robust Event Identification algorithm. As it stands now, our Event Identification algorithm only classifies footsteps and leaves all other detected events as "unclassified". A future group can create an Identification algorithm that further classifies these "unclassified" events.

If we were to tackle this problem again, we should allocate more time for researching Event Detection methods. Our group very slowly settled on kurtosis to be our underlying Event Detection model. We spent a lot of time attempting to refine two different Event Detection models until either produced promising results. Instead, if our group had spent more time researching Detection methods, it is likely that we would have began putting resources into the kurtosis model much sooner. This would have likely saved our group a lot of effort, which in turn would have given us more time to further refine our kurtosis model and begin working on a more robust Identification model.

11 Bibliography

1. Cory D. Kidd, Robert Orr, Gregory D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair MacIntyre, Elizabeth Mynatt, Thad E. Starner & Wendy Newstetter, "The Aware Home: A Living Laboratory

for Ubiquitous Computing Research", *College of Computing and GVU Center, Georgia Institute of Technology*(1999)

2. Vu, H., Nguyen, T.D., Travers, A., Venkatesh, S., Phung, D. (2017). "Energy-Based Localized Anomaly Detection in Video Surveillance." *Advances in Knowledge Discovery and Data Mining. PAKDD 2017. Lecture Notes in Computer Science()*(2017),https://doi.org/10.1007/978-3-319-57454-7_50

3. Erdal Safak, Eser Çakti. Simple Techniques to Analyze Vibration Records from Buildings. EW-SHM - 7th European Workshop on Structural Health Monitoring, *IFFSTTAR, Inria, Université de Nantes*(2014)

4. Embree, Mark. Matrix Methods for Computational Modeling and Data Analytics. https://personal.math.vt.edu/embree/cmda3606/chapter1.pdf. (2022)