Identifying the Critical Flicker-Fusion Frequency in *Drosophila melanogaster*

Kevin Gu, Jaden Singh

5/26/2020

Abstract:

The Critical Flicker-Fusion (CFF) frequency is a visual phenomenon where light flickers at a rate that is perceived as continuous. In this experiment, we aimed to establish a computational procedure that can compare action potentials -- measured as amplitude over time using an electroretinogram (ERG) -- from wild-type *Drosophila melanogaster* in order to identify the CFF frequency in a model organism. We have created a MATLAB program that uses two computational procedures (Fast-Fourier Transform comparison and Cross-Correlation measurements) to achieve this goal. Unfortunately, with the onset of COVID-19, our research ended prematurely and thus we were unable to apply the program to finding the CFF frequency. But based on visual analysis of the ERG data collected before the pandemic, we could see that the output of our program matches the trends of recorded action potential shape disparities, indicating that our code does indeed compare action potentials quantitatively. Future researchers would be able to use our program as a tool to find the CFF frequency and ultimately the difference of CFF frequency between treatment groups.

Introduction:

Scientists have long since been trying to find definitive ways to have an objective understanding of the beauty and complexities of the human brain. Research has shown that the mental state of an individual affects the frequency at which the critical flicker-fusion (CFF) occurs (Barone, 2018). This suggests that there is a possible means for scientists to use the CFF frequency as a measure for a person's neurological condition. For example, the CFF of someone diagnosed with Huntington's disease may occur at a lower frequency as the result of deteriorating nerve cells. However, in order to establish this, we would need extensive and updated research on the CFF frequency, which so far, by our preliminary research, does not exist.

Unfortunately, although it is easy to identify the CFF frequency in humans -- as we only need to ask the subjects when they are unable to discern flickers -- it is difficult and costly to obtain approval for direct experimentation on human subjects depending on the treatment. Luckily, using model organisms, specifically *Drosophila melanogaster*, we are able to quickly and easily obtain an abundance of subjects to test on that are established to be a reliable model

for human conditions. The caveat however, is that researchers lack the ability to communicate with *Drosophila melanogaster* and other model organisms in the same manner as with humans. Hence the rationale for our experiment was to be able to overcome the communication barrier by finding a way to identify the CFF frequency in a quantitative manner.

As the CFF frequency in an individual relies on retinal neurons and the action potentials they fire, electroretinogram (ERG) data was ideal for measuring and recording said action potentials as amplitude over time in wild-type *Drosophila melanogaster*. Finding the CFF frequency requires comparing action potentials of a series of light-flicker frequencies that gradually increase in increments. As a result, a crucial part of our experiment was being able to mathematically discern where the difference between action potential waveforms induced by increasing frequencies of light flicker becomes statistically significant. Creating an algorithm that is reproducible for all strains of *Drosophila melanogaster* allows other researchers to determine this frequency quantitatively, rather than subjectively deciding when the action potentials at each flicker begin to break down into irregular shapes.

We attempted this in two ways using MATLAB: comparing the raw data using cross-correlation and comparing the Fast-Fourier Transform power spectra. Both methods outputted a “similarity score” between shapes of two ERG action potentials fired from two different light flicker frequencies.

Within the scope of identifying the CFF frequency, it was reasoned that when the “similarity score” reached a significant change between a predetermined baseline light-flicker frequency (set at the lowest frequency recorded) and another light-flicker frequency along a gradient of increasing frequency, the action potential shapes at the latter frequency had broken down to a point where the fly no longer registered the light distinctively, and thus the CFF frequency would be located.

We planned on validating our method of finding the CFF using flies with Huntington's disease -- the next step in establishing the use of the CFF frequency as the measure for mental condition. Huntington's disease has been associated with visual abnormalities that may result in retinal action potentials with unique shapes when compared with control wild-type flies (O'Donnell, 2009). If we can use our established algorithm to show that there is a significant

difference between the CFF frequencies of flies modeling human neurological disorders suspected to be associated with visual abnormalities as a result of dysfunction in cortical processing, particularly Huntington's disease, we would be able to show that our method of quantifying CFF could be useful in studying human diseases (O'Donnell, 2009). Unfortunately, due to the COVID-19 pandemic, we were unable to reach this process.

Background:

Critical Flicker-Fusion Frequency:

Persistence of vision is a phenomenon which occurs in the brain by which the image produced by light that strikes the retina is retained for a fraction of a second, even after the light stimulus is removed. In humans, the image is usually retained anywhere from a fifteenth of a second to a tenth of a second (O'Donnell, 2009). Humans cannot detect discrete changes in images which occur faster than this threshold and instead perceive these changes as continuous. The name of this threshold in scientific literature is the critical flicker fusion frequency (CFF frequency), the point at which temporally spaced light stimuli begin to fuse into a perceived, continuous stimulus.

Drosophila melanogaster:

Drosophila melanogaster, commonly known as the fruit-fly is the model organism of our experiment. The organism is widely used by scientists to frequently conduct experiments as they are easy to find, easy to propagate, straightforward to approve, and cost effective. Specific to our experiment, we use fruit flies because of already established electroretinogram procedures and the abundance of available models for human neurodegenerative disorders. Although these disorders do not naturally exist in *Drosophila melanogaster* they are made possible through gene knockout and knockin techniques.

Our flies are obtained from the Bloomington Drosophila Stock Center at Indiana University Bloomington. For the first part of our experiment, we needed wild-type flies, however, the extensive catalogue at Bloomington had an entire lineup of strains. After contacting a representative, it was suggested that we use the Oregon-R-modENCODE strain (stock# 25211).

The Oregon-R-modENCODE strain is a descendant of one of the Oregon-R- lines selected by Bridges in the 1930's. It was recommended due to being Wolbachia free.

As for the validation part of our experiment, we planned to use Hungtington's disease model flies from Bloomington Drosophila Stock Center as well.

Electroretinograms:

An electroretinography (ERG) is used to measure the electrical responses of light sensitive cells from light stimulus. It produces data that can be graphed as electrical potential (measured in mV) vs. time. With this, researchers are able to see action potentials fire in real time and record them for later analysis. Our setup consists of the traditional positive and negative electrode. Applied to fruit flies, we have the negative electrode connecting the thorax of the fly to ground and the positive electrode in contact with the eye through a saline solution medium.

Researchers commonly employ ERGs to measure the CFF frequencies of model organisms by flashing an LED near or at the CFF frequency and then measuring the action potentials at corneal photoreceptors to determine, rather subjectively, when the amplitude of the initial hyperpolarization, otherwise known as the a-wave, begins to decrease significantly, and the rebounding depolarization, known as the b-wave, is immediately followed by another action potential (Wu & Wong, 1977).

For our experiment, we needed to collect action potential data in increments along a gradient from low to high light flicker frequencies. The light flicker frequency is created from the ratio between step width (the dark period measured in seconds) and pulse width (the light period measured in seconds). There are two ways to increment the frequencies: set step width and pulse width equal to each other and decrease; or keep pulse width constant at a minimal value and decrease step width. We opted for the latter due to the properties of the action potential. Since the shape of action potentials are recorded as polarization over a function of time, keeping the pulse width constant would keep the action potential shape constant on the same timespan --until it is significantly affected by the increasing frequency. Maintaining this constant was important as we were trying to determine that exact significance threshold.

Previous research conducted in the TJ Neuroscience Research Lab has looked at the critical flicker fusion frequency in *Drosophila melanogaster* as a dependent variable to determine significant differences between treatment groups. One such study, conducted in the first semester of 2017 by Ankit Agrawal, revealed that there were noticeable differences between the waveforms produced by each treatment group before and after the CFF frequency was reached, yet he provided no rigorous statistical test to show that the patterns were indeed different. Instead, he rationalized his identification of each CFF frequency by pointing out that "the signal began to spike atypically and significantly decreased in amplitude." This is an example of the subjectivity we aim to reduce by finding a quantitative way to determine the CFF.

Statistical Analysis:

To achieve the main purpose of our experiment and create an objective procedure to find the CFF, we employed two different statistical functions to compare the relativity of the recorded action potential shapes. For each fly, both methods of analysis compared every inputted file (each of which contains a series of recorded action potentials generated from a distinct light flickering frequency), to a "baseline" file containing the action potentials from a predetermined largest frequency with theoretically the most full and distinct action potential shapes.

Correlation:

In MATLAB, use of the "corrcoef" function enables the user to see how similar two signals are to each other. This method generates a correlation coefficient for two curves which are inputted as parameters by running a least-squares regression between them. The coefficient is sensitive to differences in shape and amplitude between the inputs, meaning it would theoretically allow us to locate significant changes in these characteristics as we approach the CFF threshold.

However, before running the code, it is important to align the data such that the function will produce a meaningful result. A signal generated from a higher-frequency light flicker cannot be directly compared against a lower-frequency signal since the latter would have a longer period. Capitalizing on the fact that the signals are periodic (since they are generated in response

to a light flickering at regular intervals), we decided to compare the signals using windows in between their peaks. Before running the corrcoef function, we corrected for differences in frequency by compressing the signal with the longer period so that the length of the two windows would be equal.

In this comparison, it is important to note that we chose arbitrary locations for the windows of each signal and that an arbitrary range of data was chosen for each signal (however, this range was the same for both signals throughout the entire analysis). The results we collected for this portion of the procedure represents an average of individual waveform comparisons between 22 sets of two signals, with the 10 Hz (lowest stimulus frequency) file as a standard for comparison in each one.

Fast-Fourier Transform Comparison:

In the other method of analysis, we compare the two signals using a fast fourier transform breakdown. The fast fourier transform (FFT) is a function in MATLAB which generates a vector (one-dimensional matrix) of complex doubles, each of which represent a component sinusoidal function that makes up the original periodic signal. Taking the absolute value, or magnitude, of the resultant vector generates a new vector known as the power spectrum, which can be interpreted more easily but lacks information regarding the phase shifts of the component functions.

In the plot of the power spectrum, the x-axis displays the frequencies of the component functions as bins (not Hz). The y-axis plots the amplitudes of the signals at each frequency. The actual units of the two axes in the power spectrum did not matter in our analysis as we were only comparing the relativity between the two signals.

In our FFT-based method, we created two new vectors from the two original signals which contained the frequencies of the original signals sorted in descending order based on their amplitudes in the power spectrum. As a result, the new vector for each signal contained the frequencies with the highest contribution listed at the beginning. We then created two new vectors containing the top ten frequencies from each sorted vector and compared the similarity between them using a statistical analysis technique known as “norm,” which literally returns the

distance between the two vectors in the nth dimension. Our choice to use the top ten frequencies for analysis was largely arbitrary, and the reason we did not simply compare the two full, sorted lists is because such a comparison will involve a number of frequencies which do not contribute much to the overall shape and only serve to increase the norm value, which would likely sacrifice both efficiency and accuracy. However, the process for determining a suitable size for the two new vectors warrants further investigation.

Finally, similar to the procedure for the correlation component of the analysis, we compared an arbitrary range of data that remained the same for both signals throughout the entire analysis. However, unlike the correlation analysis, we did not have to break this range of data into smaller windows to be averaged for each comparison; we could simply find the power spectrum of the entire range of data for both signals. Like the correlation method, the results are based on comparisons between 22 sets of signals, with the 10 Hz file as the standard.

Materials:

Materials for our experiment are detailed in the following list of protocols:

- "Precise Vial Preparation Protocol" written by Gundamraj and Matharoo in 2020
- "Transferring Flies Protocol" written by Gu and Singh in 2020
- "Waxing Drosophila Protocol" written by Gu and Singh in 2020
- "Electroretinogram Setup Protocol" written by Gu & Singh in 2020
- "Electroretinogram Procedure Protocol" written by Gu & Singh in 2020
- "ERG Technical Setup Protocol" written by Gu & Singh in 2020

Other materials/resources needed:

- Computer
- MATLAB

Procedure:

We first obtained wild type *Drosophila melanogaster* and followed the necessary established protocols for propagation. For preparation at the beginning of each data collection session, the fruit flies were chilled and waxed following the "Waxing Drosophila Protocol" and

the ERG was set up (fixing the electrodes, microscopes, fly positioning, etc.) with Labchart open and ready.

As explained in the background section of this paper, we changed the frequency of light flickers by gradually decreasing the step width, or "dark period," keeping the pulse width, or "light period," constant at a minimal value. In each data collection session, we generally started at a step width of 0.1 seconds and worked our way down in large increments at first until the live signal recording on Labchart began showing signs of distortion, such as a decrease in action potential depolarization and/or hyperpolarization. From there, we would decrease the value in increasingly smaller increments. Every frequency was recorded as a separate file for easy manipulation later when inputting the recorded data into our MATLAB code. Within each file, a series of action potentials were recorded as shown in Figure 1. Since we were restricted on time due to COVID-19, we were only able to complete 1 out of 15 planned trials/data collection sessions.

During the data collection session from where we pulled the data to write our program, the value we set for the pulse width was 0.005 seconds. We began with a step width of 0.09 seconds, decreasing by an increment of 0.005 seconds until we reached a step width of 0.03 seconds. From there we decreased the value by an increment of 0.001 seconds until a step width of 0.02 seconds. It is important to note that the ideal pulse width, step width range, and increments are actually 0.001 seconds, 0.01 seconds to 0.001 seconds, and 0.001 seconds respectively. These ideal values would fully utilize the capabilities of the equipment in the TJHSST Neuroscience Lab and create the full range of frequencies for an ideal data set.

Results:

The following figures provide a visual representation of the raw data as well as the two analytical methods used for comparing signals from the fly tested on March 4, our only viable trial in the experiment. We did not test the fly in a stimulus frequency range of 200-250 Hz, which is suspected to contain the flicker fusion threshold location, so we were unable to apply any methods of statistical analysis.

Although the analysis did not yield any conclusive results, trends observed in the second and third figures reveal that had we continued to stimulate the fly up to and past the assumed CFF, we may have been able to observe frequencies at and above the CFF as outliers to a fitted least-squares regression line. Since these values would likely be influential points, they may have a noticeable effect on the slope of the regression line and potentially the correlation coefficient. Otherwise, for each of the two regression lines, a residual plot could be generated with the first location of an unusually large residual value marked as the location of the CFF.

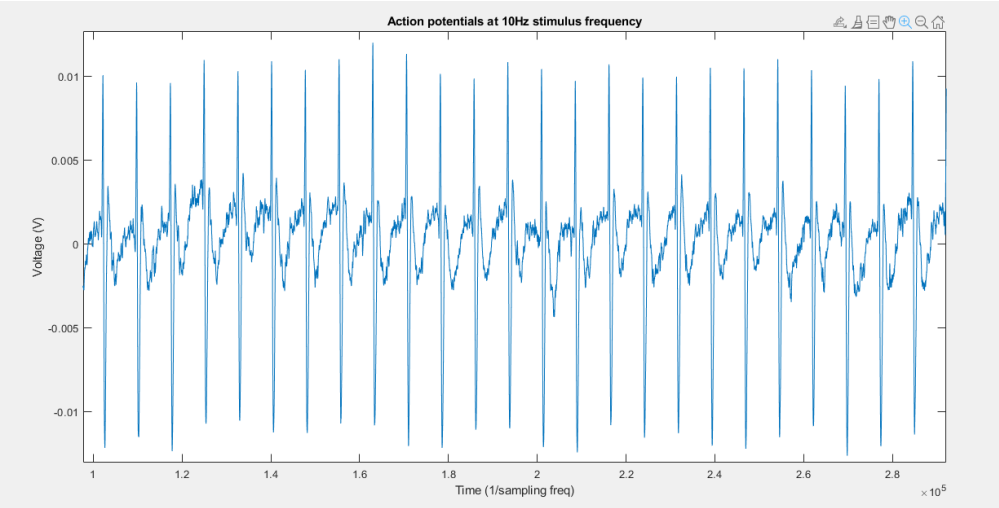

Figure 1. Raw data generated at a 10 Hz stimulus frequency. This figure shows a zoomed-in plot of the raw data collected from the LabChart software for the fly tested on March 4 after being transferred to MATLAB. The x-axis plots the time each data point was collected with an interval of the reciprocal of the sampling frequency. The y-axis plots the voltage measured at the fly's retina. About 25 action potential cycles are plotted.

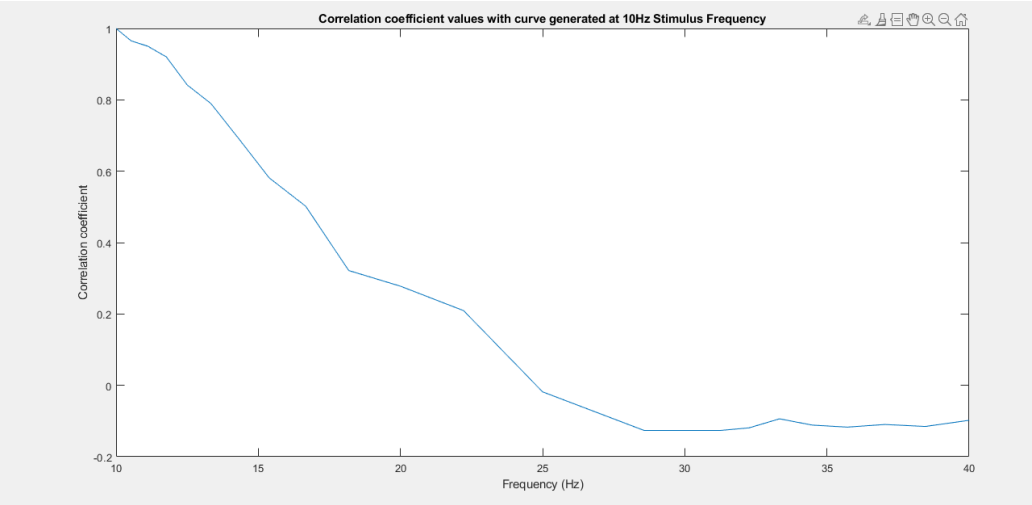

Figure 2. Correlation coefficient values with data generated at 10 Hz stimulus frequency. This figure plots the correlation coefficients generated between each file collected on March 4 and the last file collected that day, which used a 10 Hz stimulus frequency. The x-axis shows the stimulus frequency in Hz, and the y-axis plots the correlation coefficients, which can range from -1 (perfect negative correlation) to 1 (perfect positive correlation). The plot depicts a negative trend, showing that the shape of action potentials generated at higher stimulus frequencies are less similar to the action potentials generated at the lowest stimulus frequency.

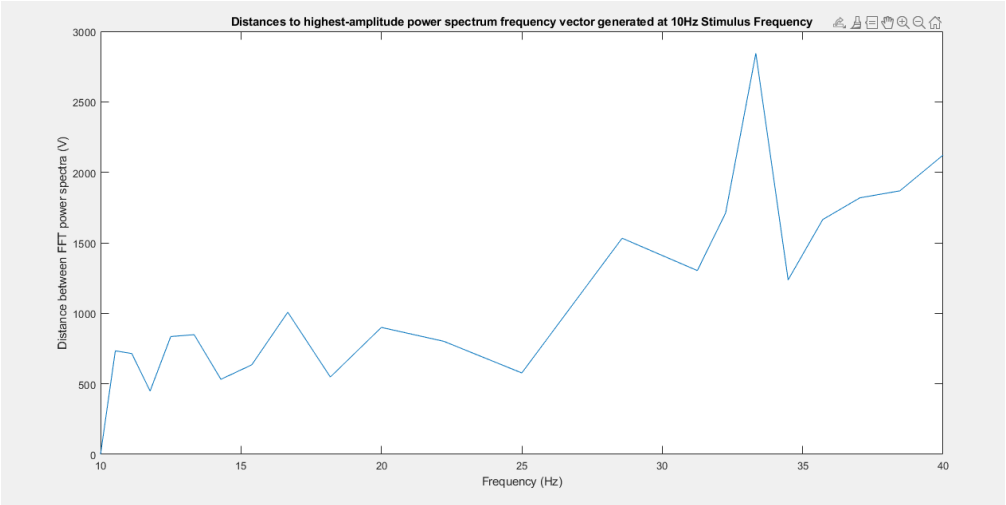

Figure 3. Distance to highest-amplitude power spectrum frequency vector generated at 10 Hz stimulus frequency. This figure plots the distances between the highest-amplitude power spectrum frequency vectors for each stimulus frequency collected on March 4 compared to the 10 Hz stimulus frequency. The x-axis plots the stimulus frequency in Hz, and the y-axis plots the distances between the vectors in V. The plot depicts a positive trend, showing that

data generated from higher stimulus frequencies have power spectra which are less similar to the lowest stimulus frequency. There appears to be an outlier at around 33.3 Hz.

Discussion:

Due to the unexpected closing of schools as a result of the COVID-19 pandemic, we were unable to collect enough data to achieve conclusive results. Before the official mandate, we were only able to collect enough viable data from four flies, since these were the only flies for which we maintained a controlled light-period and variable dark-period during the ERG procedure. Among these flies, only the fly which was tested on March 4 yielded enough data to run thorough analysis using MATLAB; even for this fly, we were unable to stimulate it at a frequency near the critical flicker fusion threshold often estimated in neuroscience literature, which is around 200 Hz (Evans, 2011).

While we were unable to conduct the 15 trials initially planned or to stimulate any flies around the often-cited CFF frequency, the agreement between the two analytical methods used warrants further investigation. For future members of the TJ neuroscience lab wishing to continue with this research, the first step should be to conduct as many trials as possible by collecting ERG data from a sufficient number of wild-type flies. However, it is also important to follow the newly-written protocols closely, as our own failure to understand the certain nuances of the procedure set us back by several weeks.

Additionally, it is necessary for future experimenters to understand how to calculate stimulus frequencies based on the pulse width and step width indicated in the LabChart user interface, as LabChart does not explicitly provide the frequencies to the user. This information was not provided in the ERG Technical Setup Protocol, so we will provide it here.

For each flash cycle, the step width indicates the period of time during which the flies remain in the dark, while the pulse width indicates the period of time during which the flies receive light. The pulse width is a constant in this experiment, so its value should never be changed throughout its duration. The stimulus frequency, in Hertz, can be calculated as the reciprocal of the sum of the two values (assuming they are in seconds) since the sum of the values gives the period of the flash cycle. For testing the CFF of *Drosophila melanogaster*, future

experimenters should beware that it is impossible to stimulate the fly at 200 Hz with any pulse width above two milliseconds, since LabChart only allows for precision of up to one millisecond and the step width must be greater than or equal to the pulse width. Additionally, due to the limitation on precision, experimenters must also be cautious of the fact that a relatively limited number of frequencies can be tested around the estimated CFF frequency of 200 Hz. Perhaps another future component of this experiment may be to find a more precise and user-friendly way to gather ERG data.

As the code for finding the similarity between ERG signals is already written in MATLAB, future research will likely build off of this code by conducting further trials which successfully reach the CFF frequency and find a method of statistical analysis which can determine the location of a significant outlier from the similarity curves. One possible way to achieve this, aso noted in our results section, is to fit a least-squares regression line to each plot and determine the first location of a major change in the slope or correlation coefficient. A residual plot may also reveal the location as a residual with an unusually large magnitude.

References:

Agrawal, A. (2018) The Effect of *Bacopa monnieri* on Locomotor Ability and Visual Responsiveness in Parkinsonian *Drosophila melanogaster*. TJHSST *Neuroscience Laboratory*.

Barone, Michele, et al. “Critical Flicker Frequency Test Predicts Overt Hepatic Encephalopathy and Survival in Patients with Liver Cirrhosis.” Digestive and Liver Disease, vol. 50, no. 5, 2018, pp. 496–500., doi:10.1016/j.dld.2018.01.133.

Evans, O., Paulk, A. C., & van Swinderen, B. (2011). An automated paradigm for Drosophila visual psychophysics. *PLoS ONE*, *6*(6). https://doi.org/10.1371/journal.pone.0021619

Lin, Z., Zhang, C., Wu, W., & Gao, X. (2007). Frequency recognition based on canonical correlation analysis for SSVEP-based BCIs. IEEE Transactions on Biomedical Engineering, 54(6), 1172–1176. doi: 10.1109/tbme.2006.889197

O'Donnell, B. F., & et al. (2009). Visual perception in prediagnostic and early stage Huntington's disease. Journal of the International Neurophysiological Society, 14, 446-453. https://doi.org/10.1017/S1355617708080405

Vanegas, M. I., Blangero, A., Galvin, J. E., Di Rocco, A., Quartarone, A., Ghilardi, M. F., & Kelly, S. P. (2019). Altered dynamics of visual contextual interactions in Parkinson's disease. NPJ Parkinson's disease, 5, 13. doi:10.1038/s41531-019-0085-5

Wu, C.-F., & Wong, F. (1977). Frequency characteristics in the visual system of drosophila. The Journal of General Physiology, 69, 705-724. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2215342/pdf/jg696705.pdf

Supplementary Materials:

Below is our code. Comments denoted with % per MATLAB syntax are made throughout for ease of understanding.

```
class CFFAnalysis
   folder=dir('C:/Users/ssene/Desktop/2019-2020 Kevin and Jaden'); %change filepath
   k=length(folder); %last file in folder as standard for comparison
   for j=50:length(folder) %change initial value (e.g. use 1 for first file)
      X=load(strcat('C:/Users/ssene/Desktop/2019-2020 Kevin and Jaden/',folder(j).name)).data(100000:600000); %change filepath/data range
      Y=load(strcat('C:/Users/ssene/Desktop/2019-2020 Kevin and Jaden/',folder(k).name)).data(100000:600000); %change filepath/data range
      xname=string(split(folder(j).name, " ")); %lines 7 to 27 find the stim frequencies for each file based on our naming system (date
stepwidth_pulsewidth.mat)
      xname=split(xname(2), ".");
      yname=string(split(folder(k).name, " "));
      yname=split(yname(2), ".");
      for a=1:2
         temp1=split(xname(1), "_");
         temp2=split(temp1(a), "-");
         for b=1:2
            xNums(a, b)=str2double(temp2(b));
         end
      end
      for a=1:2
         temp1=split(yname(1), "_");
         temp2=split(temp1(a), "-");
         for b=1:2
            yNums(a, b)=str2double(temp2(b));
         end
      end
      xfreq=1/(xNums(1, 1)/xNums(1, 2)+xNums(2, 1)/xNums(2, 2));
      Xfreqs(j-49)=xfreq; %store frequencies in a vector for plotting
      yfreq=1/(yNums(1, 1)/yNums(1, 2)+yNums(2, 1)/yNums(2, 2));

%lines 30 to 34 are for power spectrum analysis
      [~,Ix]=sort(abs(fft(X-mean(X))), 'descend');
      xindices=Ix(1:10); %take the 10 highest-amplitude frequencies for the X signal
      [~,Iy]=sort(abs(fft(Y-mean(Y))), 'descend');
      yindices=Iy(1:10); %take the 10 highest-amplitude frequencies for the Y signal
      F(j-49)=norm(xindices-yindices); %find the distance and store it in vector F

%lines 37 to 72 are for correlation analysis
      wcount=1;
      for arb=50000:10000:480000 %take multiple windows; locations are arbitrary here
         [~, xloc]=max(X(arb: arb+2*(1/xfreq)*40000));
```

```
        [~, yloc]=max(Y(arb: arb+2*(1/yfreq)*40000));
        xpk=arb+xloc; %find first peak loc for X
        ypk=arb+yloc; %find first peak loc for Y
        [~, xloc]=max(X(xpk+(1/xfreq)*40000:xpk+2*(1/xfreq)*40000));
        [~, yloc]=max(Y(ypk+(1/yfreq)*40000:ypk+2*(1/yfreq)*40000));
        xwindow=X(xpk:xpk+(1/xfreq)*40000+xloc); %create X window in between peaks
        ywindow=Y(ypk:ypk+(1/yfreq)*40000+yloc); %create Y window in between peaks
        if length(ywindow)>length(xwindow) %lines 47 to 67 compress windows as needed and find the correlation coefficients
            rscl=length(ywindow)/length(xwindow);
            count=rscl;
            for i=1:length(xwindow)
                newywindow(i)=ywindow(int32(count));
                count=count+rscl;
            end
            c=corrcoef(xwindow, newywindow);
            clear newywindow;
        elseif length(xwindow)>length(ywindow)
            rscl=length(xwindow)/length(ywindow);
            count=rscl;
            for i=1:length(ywindow)
                newxwindow(i)=xwindow(int32(count));
                count=count+rscl;
            end
            c=corrcoef(newxwindow, ywindow);
            clear newxwindow;
        else
            c=corrcoef(xwindow, ywindow);
        end
        corrs(wcount)=c(1, 2);
        wcount=wcount+1;
    end
    C(j-49)=mean(corrs); %store average of correlation coefficients for each new signal in vector C
end

plot(Xfreqs, C); %lines 74 to 78 plot the correlation curve
title('Correlation coefficient values with curve generated at 10Hz Stimulus Frequency');
xlabel('Frequency (Hz)');
ylabel('Correlation coefficient');
figure;
plot(Xfreqs, F); %lines 79 to 82 plot the FFT comparison curve
title('Distances to highest-amplitude power spectrum frequency vector generated at 10Hz Stimulus Frequency');
xlabel('Frequency (Hz)');
ylabel('Distance between FFT power spectra (V)');
```