Homework 2: Exploration of PCA algorithms.

Eunji Lindsey Lee
(*e-mail:* elee0025@uw.edu)

Abstract

We performed Principle Component Analysis (PCA) analyses on the movie files from three different cameras. We were given a total of 12 matlab files, which reflects 4 different test from three cameras each.

Contents

1 Introduction and Overview

Three videos cameras captured the oscillating (pendulum and rotating) motion of a paint can and collected data about its motion. For this assignment we will perform Principle Component Analysis (PCA) on the data to derive paint can's principal moving directions, as PCA reduces rather complex data to low dimensional and extract principal components.

This paper explores how PCA performs on 4 different types of data-data with simple oscillating motion, data with oscillation and pendulum motion and data of such motions with noises. These data changes depend on the camera shakes, orientation of cameras and the released motion of a paint can. In all four cases, the main dominating axis of a paint can is still one.

We used Python code the assignment, from extracting (x, y) coordinates of a paint can to performing PCA on the matrix, which is a collection of x and y vectors. We then did analysis of each test cases like what we had done in the previous assignment (Homework 1).

2 Theoretical Background

PCA is the reduction of data to lower dimensional to obtain main features of a data by identifying maximal variance and removing redundancies. There are 2 ways to do this:
- Use eigenvalue and eigenvector in covariance matrix
- Use SVD (Singular Vector Decomposition) on covariance matrix

As explained in the previous assignment (Homework 1), removing redundancy in the data is critical. In order to remove redundancy and to consider variance and covariance between data sets, we find the covariance matrix of a data set. The covariance matrix of n by n data matrix X is as follows:

$$C_x = \frac{1}{n-1} XX^T$$

This matrix is square and symmetric, whose diagonal terms are the variance scores and off-diagonal terms are covariance scores between all pairs of vectors. Since it is a symmetric matrix, it can be diagonalized as follows:

$$C_x = V\Lambda V^T$$

where columns of V is an eigenvector, Λ is a diagonal matrix with eigenvalues, λ_i, in a decreasing order on the diagonal.

If we now perform SVD on X, then we have the following:

$$X = U\Sigma V^T$$

where Σ is a diagonal matrix with singular values, σ_i, in a decreasing order on the diagonal. The covariance of this matrix X is then $C_x = V\Sigma U^T U\Sigma V^T/(n-1) = V\frac{\Sigma^2}{n-1}V^T$.
It shows that the two ways of performing PCA are closely related to one another, especially singular values and eigenvalues of covariance matrix via $\lambda_i = \sigma_i^2/(n-1)$.

The projections of the data on the principal axes from eigenvectors V, are called principal components. These are the new axes, which are the new efficient representation of data, and these are calculated by XV where $XV = U\Sigma V^T V = U\Sigma$.

3 Algorithm Implementation and Development

Image preprocessing was critical for this assignment as we needed an algorithm to extract positions of a moving object, a paint can for this assignment. We converted the frames into gray color and used white colors from bright white flashlight hanging on top of the paint can as well as the white color of paint can itself to find the positions of a paint can. We also used the fact that white color has the high RGB value. In order to capture whites from each image, zooming in the frame (cropping images) around moving paint can was necessary because there are some other white colors in the background such as whiteboard and the toe of shoes. Also, in order to capture white colors from a paint can more accurately, we converted to binary images such that paint can is converted to white and the rest are converted to black. When we binarize images, the range of RGB values varies depending on its brightness and blurriness. Lastly, we collected all x and y values of white colors from each binary image and calculated the average of x and y to find the final (x,y) position of a paint can from each frame.

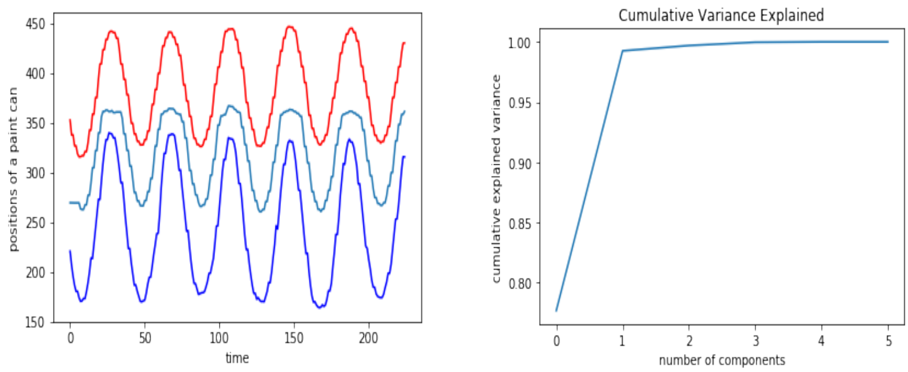
These (x,y) coordinates of a paint can from each frame are combined into a vector for each camera. Note that these (x,y) coordinates are 2-D representation of a system by cameras, but these are not the actual $x-y$ plane of the system.

Since a capture of oscillating motion by cameras were not synchronized, we plotted the graphs of each (x,y) coordinate vectors versus time into one single graph and adjusted data appropriately to make wave oscillations to be in phase and lengths of these vector (say n) to be identical. In order to use PCA, we combined these vectors x and y carefully into a single matrix which is in the following order:
x vector from camera 1
y vector from camera 1
x vector from camera 2
y vector from camera 2
x vector from camera 3
y vector from camera 3
such that the matrix is in $\mathbb{R}^{6xn}$. Centering the data by subtracting the matrix by its mean is significant to obtain the correct covariance matrix. If not covariance matrix would not be equal to $C_x = \frac{1}{n-1}XX^T$.

4 Computational Results

4.1 (test 1) Ideal case

The entire motion of this case is in the z-directions with simple harmonic motion to be observed. Before we do PCA analysis, we loaded data from 3 mat files (cam1_1.mat, cam2_1.mat and cam3_1.mat). After extracting (x,y) coordinates as described in the previous section, we plotted and aligned waves in phase as shown in figure 2 part (a).



(a) (b)

Fig. 2: (a) shows the plot of paint can positions by three cameras in phase and (b) shows the plot of cumulative explained variance versus the number of components

Figure 2 part (b) shows that one single component captures almost all of the energy. That is, test 1 can be explained by one single principal component or principal direction. This result was what we expected to obtain since the z-direction is the single direction that explains the entire motion. Figure 3 supports this argument as only one significant component shows the simple harmonic oscillation.

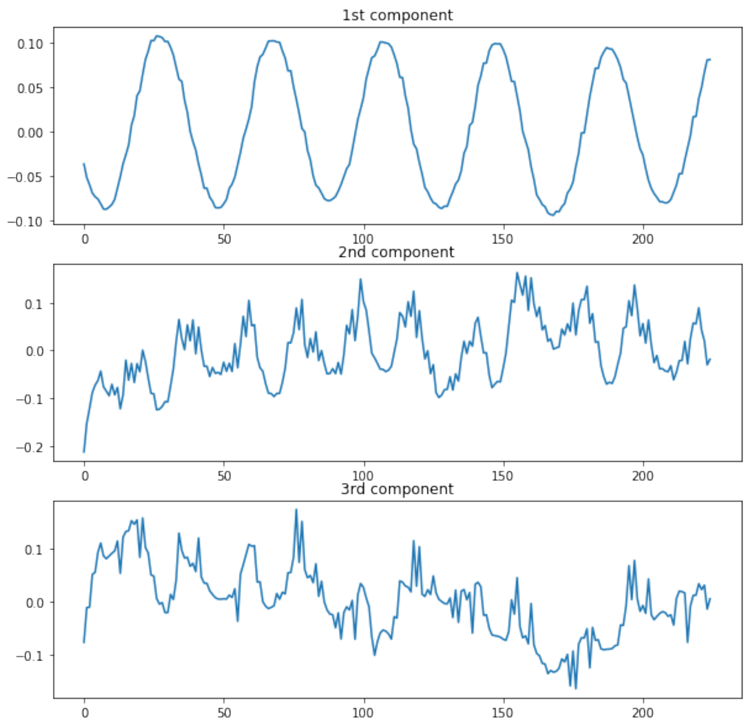
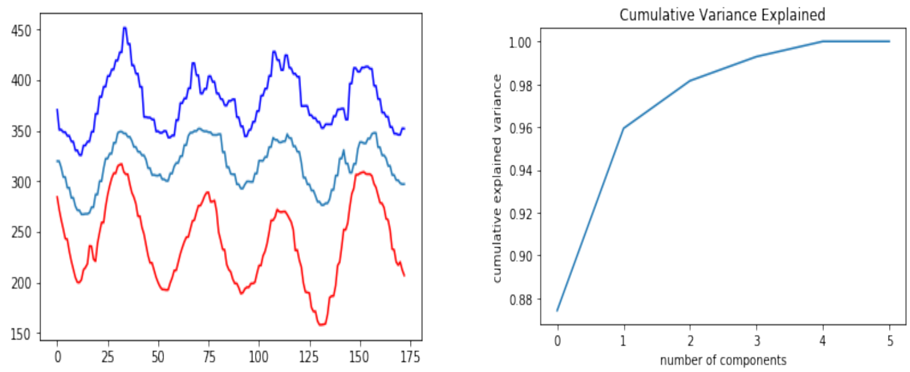


Fig. 3: three main components

4.2 (test 2) noisy case

Test 2 is similar to the ideal case except that now it includes camera shakes, which is considered as noises in the data. Similarly for test 2, before we do PCA analysis, we loaded data from 3 mat files (cam1_2.mat, cam2_2.mat and cam3_2.mat). Then plotted the aligned waves in phase as shown in figure 5 part (a)



(a) (b)

Fig. 5: (a) shows the plot of paint can positions by three cameras in phase and (b) shows the plot of cumulative explained variance versus the number of components

The plot from figure 5 (a) is not as smooth as test 1 and more principal components are required to capture most of the energy. It means more directions are required to explain the motion for test 2. However, as seen in figure 6, despite the existence of noises, only first component reflects the simple harmonic oscillation and the other two components do not reflect any certain patterns. Moreover, as seen in 5 (b), first component captures 96 percent of the data so it is safe to assume that one principal component captures most of the dynamics of the objects movement. And again, this corresponds to what we expected from the data as it is similar to the ideal case experiment.

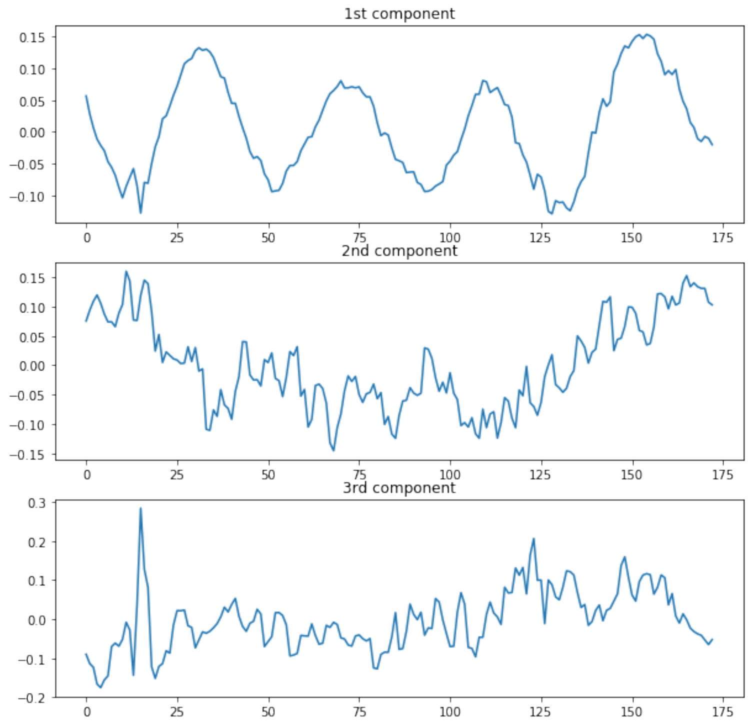
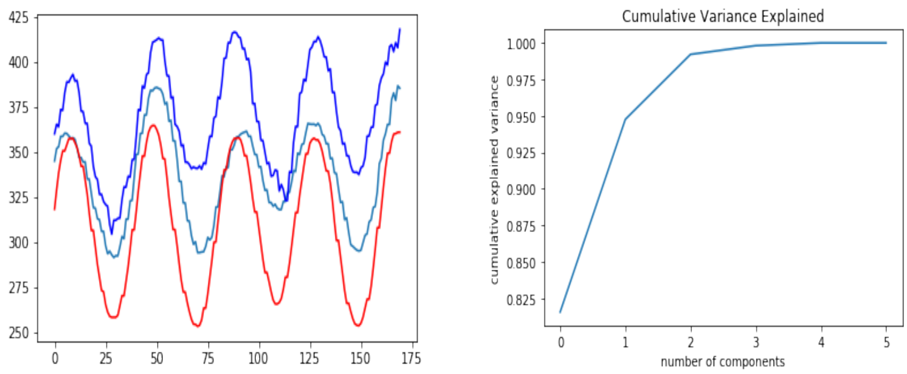


Fig. 6: three main components

4.3 (test 3) horizontal displacement

In this case, the mass is released off-center so as to produce motion in the *xy* plane as well as the z direction. Thus there is both a pendulum motion and a simple harmonic oscillations. To observe this, load data from 3 mat files (cam1_3.mat, cam2_3.mat and cam3_3.mat) as done for previous test cases. Then, similarly, plotted the aligned waves in phase as shown in figure 8 part (a).



(a) (b)

Fig. 8: (a) shows the plot of paint can positions by three cameras in phase and (b) shows the plot of cumulative explained variance versus the number of components

The plot from figure 8 part (a) is fairly smooth and figure 8 part (b) shows that 2 main components captures almost all of the dynamics of the motion. It's also supported by figure 9, as only first two components reflects the wave motion. This supports our expectation as test 3 is explained by both a pendulum motion and a simple harmonic motion.

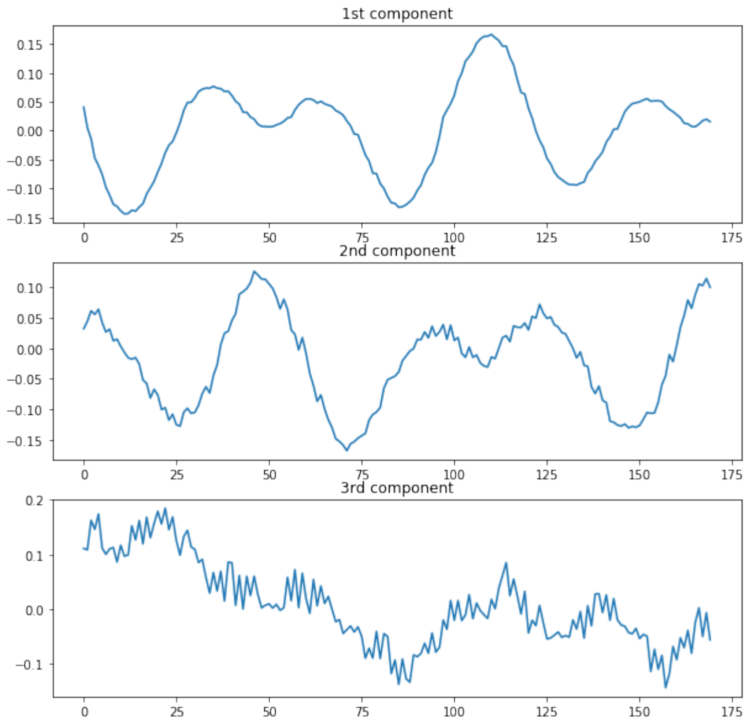
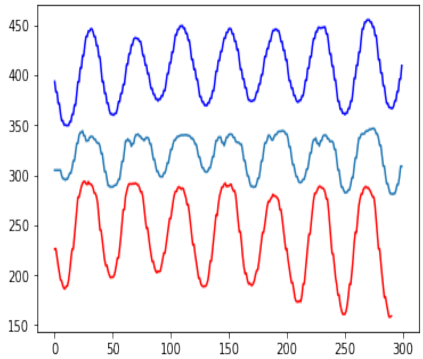


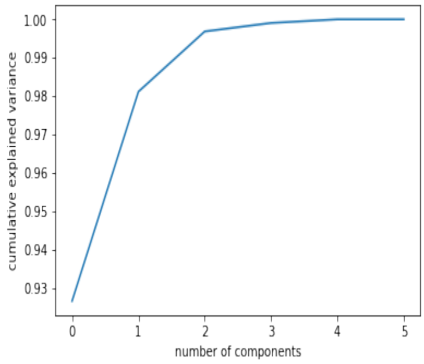
Fig. 9: three main components

4.4 (test 4) horizontal displacement and rotation

In this case, the mass is released off-center and rotates so there is both a pendulum motion and a simple harmonic oscillations. To start off with this test case, again, load data from 3 mat files (cam1_4.mat, cam2_4.mat and cam3_4.mat). Also, plotted the aligned waves as done for previous test cases.



(a)



(b)

Fig. 11: (a) shows the plot of paint can positions by three cameras in phase and (b) shows the plot of cumulative explained variance versus the number of components

The plot obtained from figure 11 part (a) shows a fairly clean and smooth wave motion. Similar to test 2, as shown in figure 11 part (b), two main components captures almost all of the dynamics of the motion. Furthermore, it is also supported by figure 12. The result was expected as the test 4 was explained by 2 main motions (both a pendulum motion and a simple harmonic motion).

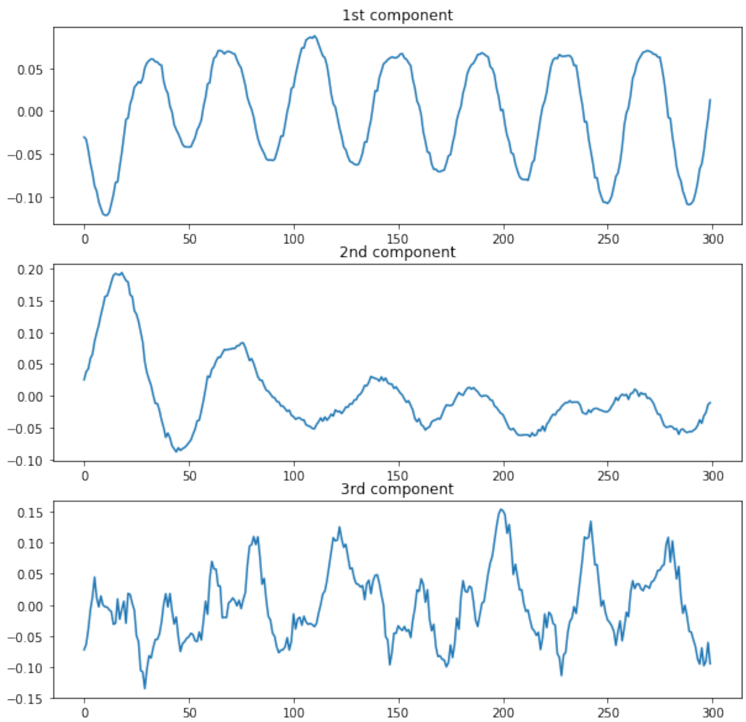


Fig. 12: three main components

5 Summary and Conclusions

Taking coordinates from the paint can was the hardest part of this assignment, which is similar to what machine learning is like in real life, as the hardest part of the most analyses is to pre-precess the data. Despite its hardship, the result for the following 4 test cases were satisfactory as PCA analyses reflects what was expected from their results. However, the result would be more successful if we have the method to clean the data in order to reflect more smooth wave motion, thus obtain the better principal components of each test cases.

6 Appendix

Link to the a python file and a html file:
https://drive.google.com/open?id=1fmgabwpYUbME2QTHo8DlCAQWMAr420Em

References

AMATH 582 Textbook. J. Nathan Kutz. 2013. Data-Driven Modeling Scientific Computation: Methods for Complex Systems Big Data. Oxford University Press, Inc., New York, NY, USA.