Studying Simple Harmonic Motion with Principal Component Analysis

Crystal Yang

ABSTRACT

In this article, I am going to use the videos of dangling a mass to study the mass's motion. I am going to extract the location information of this mass by MATLAB, perform Principal Component Analysis (PCA) by using the Singular Value Decomposition (SVD) on the dataset and interpret the result to study the simple harmonic motion of it.

INTRODUCTION

To visualize data with multiple variables is hard, and usually people are interested in finding the dominant variable that driving the system, that is, usually there is multiple variables may be measuring the same thing, so they are moving in the same direction, so we want to reduce the complexity in the data by using the SVD.

We have three cameras recording the same dangling motion of a mass with different shooting angle, so we can extract 6 variables each indicating the x-y coordinate of the video. These 6 variables consist of the dataset that we want to study about. The mass is mainly dangling (1-directional movement), but we have four situations with each we introduce different level of disturbance to make it comparable with each other: 1) The camera is stable, and there is no noise or disturbance, and this is the ideal situation; 2) The camera is shaking, makes it looks like that the mass is shaking; 3) The mass is released off-center so it has some horizontal movement instead of purely dangling up and down; 4) The mass is released off-center so it is rotating and dangling up and down. We are interested in interpreting and comparing the PCA result from the x-y coordinate information collected from these videos to see what kind of motion the system is performing.

THEORETICAL BACKGROUND

The Singular Value Decomposition is a factorization of a data matrix and splits the matrix into three parts by finding the eigenvectors of the matrix through polar decomposition. Given a matrix X, we can decompose it into three matrices $X = U\Sigma W^T$ where U is a matrix with orthonormal columns, and Σ is a diagonal matrix that contains the singular value that describe the importance of the dimensions, and W^T forms an orthonormal basis that spans the dimension. Now let us connect this with the Principal Component.

Imagine we have a matrix X with multiple vectors, we can compute the variance and covariance of this matrix by $\frac{1}{n-1}XX^T$ and denoted it as C_X:

$$X = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}; \qquad \frac{1}{n-1} XX^T = \begin{bmatrix} \sigma_a^2 & \sigma_{ab}^2 & \sigma_{ac}^2 & \sigma_{ad}^2 \\ \sigma_{ba}^2 & \sigma_b^2 & \sigma_{bc}^2 & \sigma_{bd}^2 \\ \sigma_{ca}^2 & \sigma_{cb}^2 & \sigma_c^2 & \sigma_{cd}^2 \\ \sigma_{da}^2 & \sigma_{db}^2 & \sigma_{dc}^2 & \sigma_d^2 \end{bmatrix}$$

We want to find a new set of coordinates so that the variables are uncorrelated (find a way to express all the variables), we want to compute the eigenvectors of the covariance matrix C_X as the unit vector of these eigenvectors tell us by how much the data span a dimension. Thus, we want to diagonalize C_X to make it easier to compute the eigenvectors, so we got:

$$C_X = V\Lambda V^T$$

where V is the eigenvector matrix, and the eigenvectors are uncorrelated and orthogonal to each other since C_X is a symmetrical matrix.

PCA is associated with SVD, consider a matrix A:

$$A = \frac{1}{\sqrt{n-1}} X; \qquad then\ C_X = \frac{1}{n-1} XX^T = AA^T$$

Then we can express that in SVD:

$$C_X = AA^T = U\Sigma^2 U^T$$

So, the square singular value is the eigenvalue of the covariance matrix. To use the change of basis in PCA, we use the property that U is an orthogonal matrix so that its inverse equals to its transpose, we got the new coordinates as:

$$Y = U^T X$$

$which\ has\ a\ covariance\ matrix\ \frac{1}{n-1} YY^T = \frac{1}{n-1} U^T XX^T U = U^T AA^T U = U^T U\Sigma^2 U^T \mathrm{U} = \Sigma^2$ and proves that the variables in Y are uncorrelated since $\Sigma^2$is a diagonal matrix.

Thus, to find out the principal component of from the SVD, we need to look at the singular value of the SVD result. The larger singular value indicates heavier contribution this variable has toward the whole system, and we can extract that variable as the principal component of the system.

ALGORITHM IMPLEMENTATION

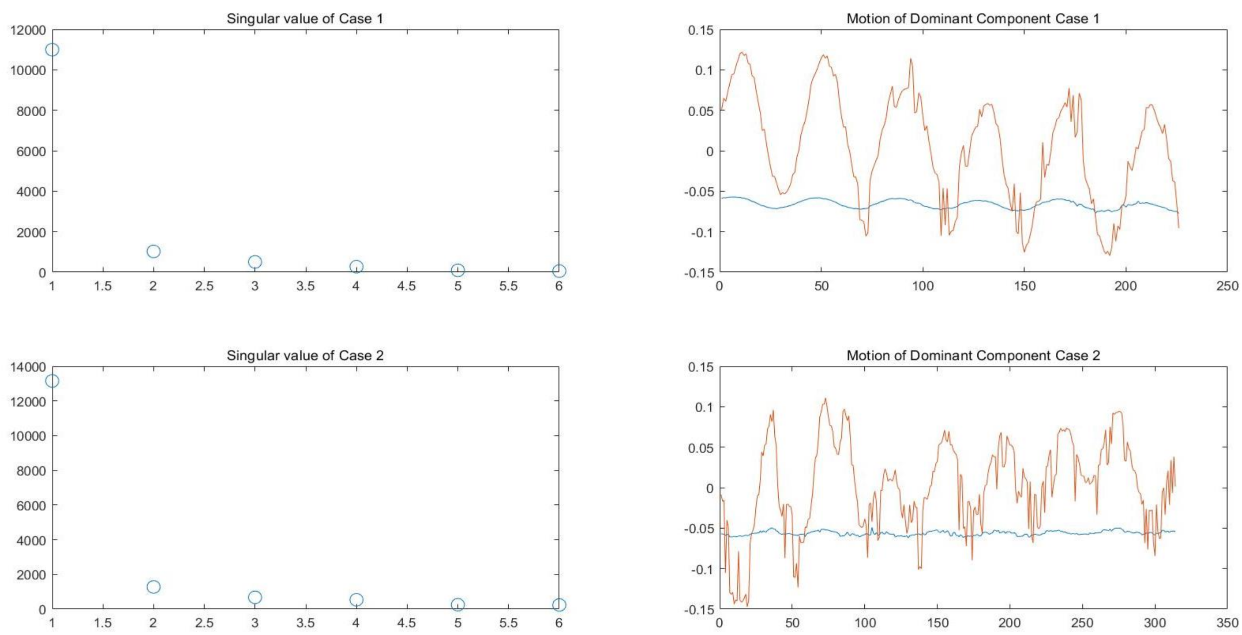
First, to apply the above analysis to see the result of the motions, we need to extract the mass positions from the videos by the three different cameras. Loading the data into MATLAB, I am going to locate the position of the mass by looking at every frame of the videos, which reduce the problem of finding the position of one pixel in the graph.

To do this, we put a light source on the mass, so we can use it and trying to locate the brightest pixel in each frame. Given one frame from the video, I turned the colorful graph into a grayscale graph using the `rgb2gray()` function in MATLAB so the pixel is only containing the shade of gray in a range of (0,256) different shade of gray from black to white, this makes us easier to track the bright light source which contains mainly white. Thus, we can use the `max()` function to find the pixel that is brightest, and storing the pixel's x-y coordinates information. Note that there is white wall behind the mass, so we need to portion the frame so that we will not accidentally capture the position of the background. I done this by observing the movement of the mass and choose the suited size to look at accordingly.

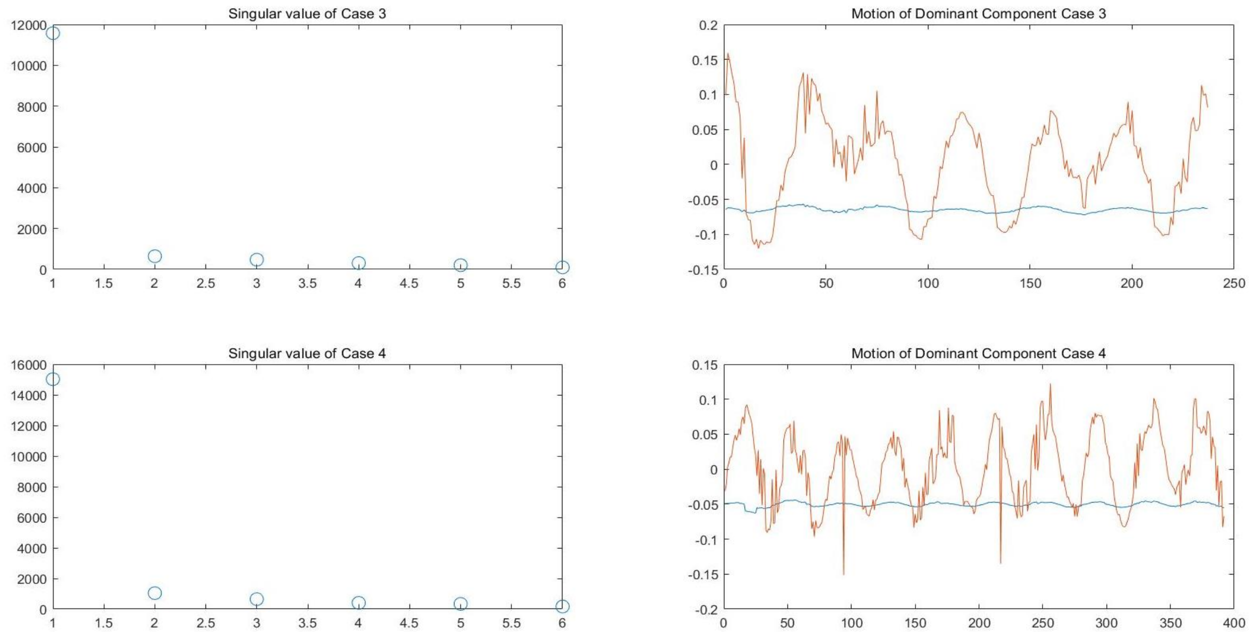
For each camera, we will have 2 vectors separately containing the x coordinate information and y coordinate information, thus, for 3 cameras, we will have in total 6 dimension of data to consider with, and we will aggregate these 6 dimension of data into one matrix and then use `svd()` function in MATLAB to extract the singular value matrix. With each video has different length, I will trim the video according to the shortest videos to make sure that they are in the same length and the mass is moving together for all the videos.

COMPUTATIONAL RESULTS

In this section, I am going to show the resulting principal component of the 4 cases by looking at the singular value matrix from SVD and selecting the principal components that contribute the most to the model, contrast and explain any differences causing by the disturbance we added:



Case 1 and Case 2 PCA 1



Case 3 and Case 4 PCA 1

From the above graphs that show the singular value of each case's SVD result, we can see that always the first variable and second variables in the model is always dominating the change of the system, and as the rest of the variable's singular value fast decreasing to zero, we can see that the complexity of the original 6 dimensions dataset can be reduced to 2-dimension dataset where we only include the first two elements. By looking at the plot of the first two columns of the V matrix from the SVD, we can clearly see that the mass is doing harmonic motion from the sinusoidal movement.

For Case 1, this is the ideal case where we see this perfect sinusoidal movement captured by the first and second variables.

For Case 2, because of the shake of the camera, we are not able to trace the movement of the mass perfectly, thus there are a lot of bumps in the plot and it is hard to see the sinusoidal shape of the plot, but from the plot of the dominant variables (the first two variables), we can still recognize the shape of this periodic movement.

For Case 3, as the mass is releasing off the center, at first the mass is move horizontal in another direction (z direction), thus, we can see that the up and down movement is not that quit obvious compare with Case 1.

For Case 4, we can see that there are more peaks than in Case 1 which means that the period of one up and down decrease. This is due to the rotation we had when release the mass off center, it miss-recognize the rotation's height change as x-y coordinate change, thus, resulting in shorter ups and downs.

Because we are doing this experiment repeatedly, the max height of the mass release from the center is not guaranteed to be the same each time, thus, we can see the amplitude of the periodic movement is not the same each time.

SUMMARY & COMCLUSION

In summary, we practiced the SVD on studying the harmonic motion of a dangling mass, and we can conclude that given 6 dimensions of data, we are able to reduce the dimension of data into 2 to describe the system that contains the moving mass even if we have different level of disturbance when observing the data. We have shown that the Singular Value Decompositions is a power tool to efficiently and accurately reduce the complexity exist in the dataset to allow simplification in further analysis.

Appendix A. MATLAB functions used and brief implementation explanation

`Rbg2gray():` takes an true color RBG image and returns a grayscale image.

`svd(X, ‘econ’):` takes an matrix X, returns a reduced Singular Value Decomposition of the m-by-n matrix X, that is, the returned U only contained n columns, S is n-by-n, and only the first m columns of V are computed. This reduced form of SVD is sufficient for our explanation.

Appendix B. MATLAB Codes

```
% Loading the videos
load('cam3_1.mat')

numFrames = size(vidFrames3_1,4);
for j = 1:numFrames
    X = vidFrames3_1(:,:,:,j); % just capture one frame of the graph
    gray = rgb2gray(X);
%     por = gray(:,320:400); % for cam1
%     por = gray(:,200:300); % for cam2
    por = gray(200:400,200:end); % for cam3
    [~,idx] = max(por(:));
    [row,col] = ind2sub(size(por),idx);
    x_3_1(j) = col+200;
    y_3_1(j) = row+200;
% Use the following code to check whether the point is on the object
%     imshow(X); drawnow,hold on
%     plot(200+col,200+row,'r.');
%     pause(0.5);
End

%% trim the dimension to be the same
x_2_1 = x_2_1(50:end-9);
y_2_1 = y_2_1(50:end-9);
x_3_1 = x_3_1(7:end);
y_3_1 = y_3_1(7:end);
plot(1:226,y_1_1,"m-"), hold on
plot(1:226, y_2_1,"k-")
plot(1:226,y_3_1,"g-")

%% aggregate into one matrix to apply pca
d_1 = [x_1_1;y_1_1;x_2_1;y_2_1;x_3_1;y_3_1];
[U_1,S_1,V_1] = svd(d_1,'econ');




%% PLOT the SVD
figure(1)
subplot(2,2,1)
plot(diag(S_1),'o',"MarkerSize",10);
title("Singular value of Case 1")
subplot(2,2,2)
plot(V_1(:,1:2));
title("Motion of Dominant Component Case 1")
subplot(2,2,3)
plot(diag(S_2),'o',"MarkerSize",10);
title("Singular value of Case 2")
```

```
subplot(2,2,4)
plot(V_2(:,1:2));
title("Motion of Dominant Component Case 2")
figure(2)
subplot(2,2,1)
plot(diag(S_3),'o',"MarkerSize",10);
title("Singular value of Case 3")
subplot(2,2,2)
plot(V_3(:,1:2));
title("Motion of Dominant Component Case 3")
subplot(2,2,3)
plot(diag(S_4),'o',"MarkerSize",10);
title("Singular value of Case 4")
subplot(2,2,4)
plot(V_4(:,1:2));
title("Motion of Dominant Component Case 4")
```