# Principle Component Analysis: Recorded Spring-Mass System

Zhaokai Wu, Department of Mechanical Engineering, University of Washington


## Abstract

Singular value decomposition (SVD) is one of the most power technique in data analysis. SVD is a universal tool in feature extraction by reducing the sample space into a feature space that can be represented using very few feature vectors, namely principle components. The importance of each principle components can be determined through mode analysis, a technique using singular values of a factor of data matrix. In this project, we explore using SVD in analyzing a mass-spring system recorded by 3 cameras at difference locations. We segmented out the effective movement region by reconstructing the modes into images. Finally, we tracked the motion of the mass in various movement patterns.


## *Sec. I. Introduction and Overview*

In data analysis, large scale data is often collected and stored in a matrix form. In real world application, theses data are complex. The goal of data analysis is to extract meaningful information and turns it into insights. For instance, we could extract the Newton Second Law by collecting digital data from cameras at different position filming a mass-spring system. Obviously, the massive image data embedded in a data matrix captures the oscillation movements. However, the redundancies and noises entangled in the data sets are obtrusive. Starring at an unreasonably large chunk of matrix can get us nowhere near having the same euphoric appreciation on the elegantly compact formula

$$F = m \cdot a$$

Fortunately, in the era of data there are techniques so elegant too that we can use on these data matrix and produce undiscovered knowledge that never be mined by a human brain. It takes a leap of imagination and some linear algebra knowledge to think about these data lives in an abstract space, namely the sample space, where there exists some fundamental molecule that by freely combining(addition) or scaling (scalar multiplication) can form every piece of initial unprocessed data in the matrix. One of the most powerful tool in data analysis is called singular value decomposition (SVD). It guarantees to condense any form of data matrix by transforming the sample space into a feature space which normally can be described by very few elements. They are also called the bases of the feature space. The process of this condensation of key information is often referred as dimension reduction. Among bases of feature space, there are always some more dominant than others. Usually, the dominant bases are the attributes we as data analysts aim to mine. When data collection is done correctly, the insignificant bases are often subtle and contains noise information. We use technique called mode analysis to decide to keep or truncate certain molecule.

The data we analyzed in this experiment is consisted of four sets videos record a mass-spring system in four tests:

- (test 1) Ideal case: Consider a small displacement of the mass in the z direction and the ensuing oscillations. In this case, the entire motion is in the z directions with simple harmonic motion being observed (camN 1.mat where N=1,2,3).
- (test 2) noisy case: Repeat the ideal case experiment, but this time, introduce camera shake into the video recording. This should make it more difficult to extract the simple harmonic motion.

But if the shake isn't too bad, the dynamics will still be extracted with the PCA algorithms. (camN 2.mat where N=1,2,3)

- (test 3) horizontal displacement: In this case, the mass is released off-center so as to produce motion in the x−y plane as well as the z direction. Thus there is both a pendulum motion and a simple harmonic oscillations. See what the PCA tells us about the system. (camN 3.mat where N=1,2,3)
- (test 4) horizontal displacement and rotation: In this case, the mass is released off-center and rotates so as to produce motion in the x−y plane, rotation as well as the z direction. Thus there is both a pendulum motion and a simple harmonic oscillations. See what the PCA tells us about the system. (camN 4.mat where N=1,2,3)

The data obtained by cameras are compacted in matrix, which we applied SVD on. Then we tracked the motion of the mass by segmenting out the effective movement region and indented the brightest pixel in each frame.

## Sec. II. Theoretical Background

1. Statement of the Theorem

For arbitrary matrix $M$ with shape $m \times n$ from filed $\mathbb{F}$, where $\mathbb{F}$ represents either real or complex field, there always exists a factorization, called a singular value decomposition of $M$ in form

$$\mathrm{M} = U\Sigma\mathrm{V}^*$$

where U is $m \times m$ unitary matrix; $\Sigma$ is a diagonal matrix shaped in $m \times n$, where the non-negative diagonal entries $\sigma_i$ are referred as singular values of $M$; $V^*$is a $n \times n$ unitary matrix.

2. Geometric Interpretation

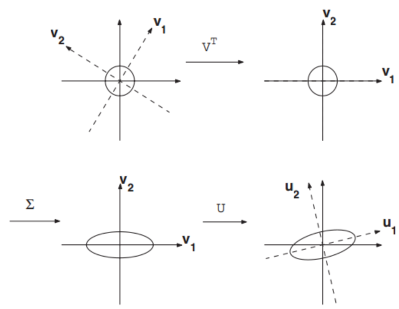



figure 1. Geometric meaning of SVD: The image of a circle under matrix multiplication is a ellipse

A high-level observation to make about the SVD theory is that any linear transformation can be viewed as a composition of rotation or reflection, a scaling, and another rotation or reflection. To illustrate this geometric interpretation, we will provide a demonstrates the SVD decomposition of a matrix multiplication on a 2D circle. Then, we claim that this geometric interpretation can be extended to any dimensions.

Considering the three factors of the SVD separately, note that $V^*$ is a pure rotation of the circle. Figure 1 shows how the axes $v1, v2$ are first rotated by $V^*$to coincide with the coordinate axes. Secondly, the circle is stretched by $\Sigma$ in the directions of the coordinate axis to form an ellipse. Lastly, the ellipse is rotated by $U$ into its final position. Note how $v1, v2$ are rotated to end up as $u1, u2$, the principal axes of the final ellipse. A direct calculation shows that $Mv_j = \sigma_j u_j$. In other words, $v_j$ is first rotated to coincide with the $j$ coordinate axis, stretched by a factor $\sigma_j$ , and then rotated to point in the direction of $u_j$.

It is not to illustrate this composition of transformation is also applicable in 3D space, where the object of examination is replaced with a unit circle. However, as human we are limited to construct our imagination around objects that's less or equal to three dimension. Therefore, it is only through the power of mathematics, particularly linear algebra, that we reached the conclusion that this decomposition is also true in higher dimension.

3. Singular Value and Mode Analysis

Singular values are contained in matrix $\Sigma$. It determines the magnitude of stretching in the transformed sample space $V^*$. In the example of a 2D unit circle, the largest singular value is the length of the longest principle axis. In high dimension data analysis, $\Sigma$ stretches the bases of $V^*$. If $M$ is normalized and zero centered with mean, singular values can be interpreted as the importance of each base vector. In other words, if some singular values are significantly smaller than others, their contribution to the characteristics of the data is minimal and often neglectable. The technique to distinguish the importance of the bases and selectively neglect unimportant information (often turns out to be noise) is a called mode analysis.

## ***Sec. III. Algorithm Implementation and Development***

1. Loading Videos

It is useful to think about videos are composed of series of images arranged in a timely order. Since the videos are given in a preprocessed MATLAB array "*.mat" files, We used the "scipy.io" module to directly read and store each videos in a numpy array. Note that each video array has four dimensions

$$video\ array = [IM_{HEIGHT} \quad IM_{LENGTH} \quad RGB\ Channels\ No. \quad Frame\ No.]$$

where each frame has a resolution $480 \times 640$ and there are around 250-300 frames per videos. We unified the length of all video by keeping only the first 226 frames.

2. Playing Videos

It is necessary to study the videos from a human intuition perspective to gain a better idea of the system with different movement of mass. Therefore, we used "animation" module in matplotlib to reconstruct the videos from arrays. Figure 1 shows a screen shot of video we reconstructed. Notice we used all three-color channels in video reconstruction.


*figure 1: A example screen shot of one video loaded on jupyter notebook as a html5 video format*

3. <u>Applying SVD</u>

Since each test is recorded with three cameras from different angles, we tried out couple data compaction method. The first boxing method we applied is to compact all data collected from three cameras in one matrix as:

$$\begin{bmatrix} cam1\_frame1 & cam1\_frame2 & \dots & cam3\_fram226 \\ cam2\_frame1 & cam2\_frame2 & \dots & cam3\_fram226 \\ cam3\_fram1 & cam2\_frame2 & \dots & cam3\_fram226 \end{bmatrix}$$

The second boxing method was done on individual cameras, for instance:

$$[cam1\_frame1 \quad cam1\_frame2 \quad \dots \quad cam3\_fram226]$$

As for the SVD tool, we used the Scipy Sparse SVD to calculate the first 100 modes of each video. It generally takes up to one minute to compute a matrix boxed in method 2.

4. <u>Image/Video Reconstruction</u>

The principle directions contained in the feature space matrices obtained by SVD method were reconstructed as grey scale images by reshaping into the image resolution and print our using pyplot.

5. <u>Motion Tracking</u>

One of the challenges we encounter was to segment out the moving mass from the surroundings in all frames. From figure 1, we can identify some unwanted features: people, chair, table, etc. Since the people in the videos are not stall, applying SVD directly on all frames will result in some unwanted feature vectors, therefore we decided to manually crop out the unwanted features. We did so by reconstructed the most important mode in a grey scale image and read off the range of region of interest.

One of the interesting task we completed was to track the motion of the mass by taking advantage of the fact that there was a flash light installed on the mass. The process flow is as following. Firstly, we identified the surrounding by printing the reconstructed image of mode 1. For instance, we picked range [200,400] in x and range [ 200,400] in y. After cropping, images will look something similar like figure3.

After cropping, we identified the position of the flash light in each frame by calling the pixel with largest intensity and obtained its x,y position. We plotted the x,y position with respect to time.

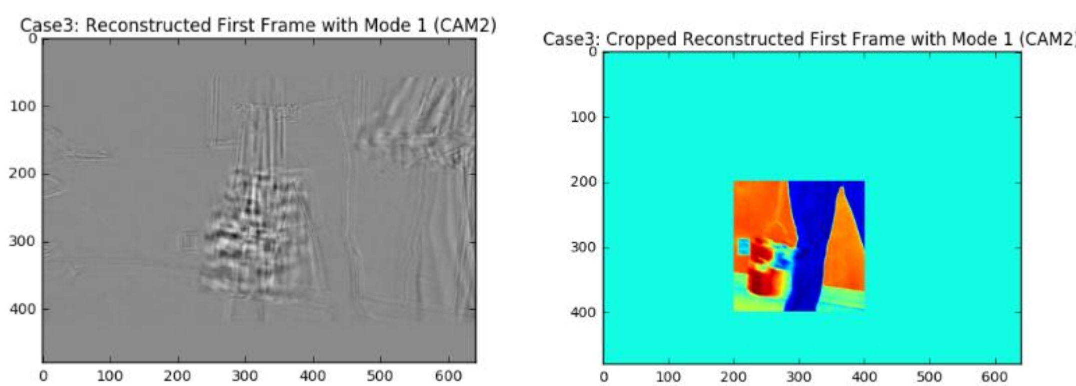



figure2: Case3 reconstructed of 1st principle mode with camera2 (first frame) (left)

figure3: Case3 cropped reconstructed 1st principle mode with camera2 (first frame) (right)

## Sec. IV. Computational Results

### 1. Boxing Method 1

Boxing information with method 1 form three cameras returns an unideal smoothly decayed singular value curve. In other words, each feature space vector has similar significance in terms of representing the sample space. Figure 4 shows the modes distribution for test 2. All three other tests return similar distribution when data is compacted in one matrix. These results will not give us any insights so we turned to boxing method 2 in investigating individual cameras. This makes the task of dimension reduction difficult science we don't know how to truncate the modes. Without going too in depth for pursing explanations for this phenomenon, we pose one hypothesis that can potentially contribute to the behavior. We think the videos might be out of phase of each other. In other words, these three cameras are not recording the system simultaneously.

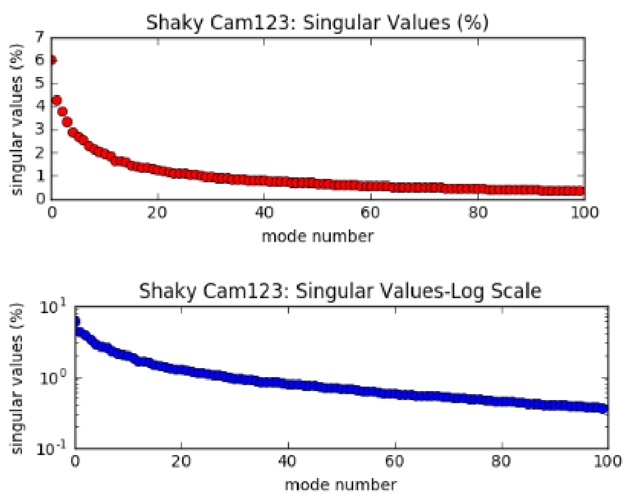
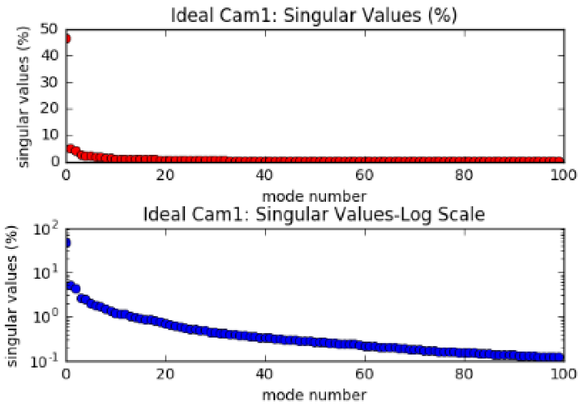



*figure 4: singular value distribution for test 2*

### 2. Boxing Method 2

#### 2a. Test 1&2

For the first test, camera 1 captures the movement of the mass nicely and the SVD results turned out to match out intuition about the system with one major mode representing the mass moving up and down. The second test records the same dynamic system with one mass moving vertically in space. The only difference is that the recorded shakes the camera while recording, which introduces noises. SVD turns out to still can pick out the mass movement. The only difference is that the singular value of the first mode drops from ~47% to ~28%. The intuition is that, the smaller the magnitude of the modes, the less "confident" of the SVD method is about the feature extraction. However, even with decreased "certainty" about extraction, we are still able to pick out one major mode. The result is impressive in a way that it is very hard for a human eye distinguish the moving mass from the noisy recording but SVD does a nice job.



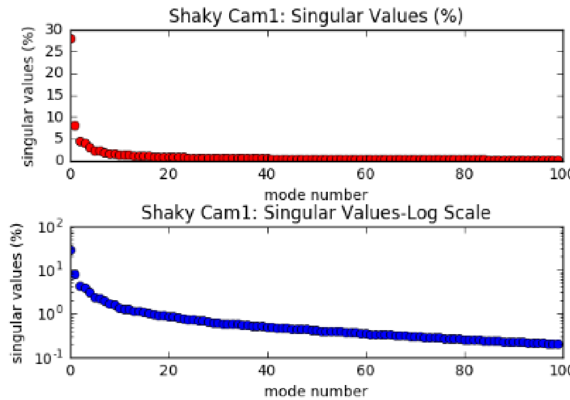




*figure 5: singular value distribution for test 1(left); figure 6: singular value distribution for test 1(right)*

**2b. Test 3&4**

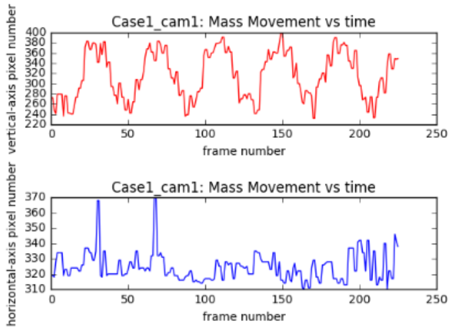
We applied to SVD on data matrices of camera 1,2,3. Again, 1 or 2 modes stand out of each matrix. To compare the relative significance of all 6 modes. We compared the magnitude. The singular values are 32, 33, 45, 45, 39, 39 respectively. We arrive to a conclusion that camera 2 represents the best of the system since it has higher singular values. In other words, there exist a best point of view for viewing the system and for test 3 it's the point of view of camera2. Figure 2 also shows empirically cam 2 is the best data matrix to use for motion tack. Like test 3, we calculated that camera 2 also gives best data that represents the system.

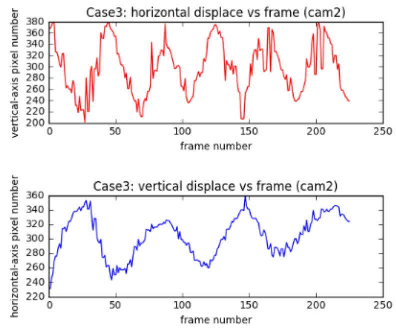
## 3. Motion Capture

Our algorithm tracks the motion of the mass in vertical and horizontal position. Figure 7 shows the vertical and horizontal movement of the brightest pixel in time. We can clearly identify the oscillation of in the upper plot. On the other hand, the horizontal movement in test 1 makes less sense. The two spikes occurred probably due to error introduced by manually cropping. An extra piece of information comes from computing the variance of both displacements. It turns out that the horizontal movement is neglectable. While the vertical displacement has a variance of 2188.5, the horizontal displacement has a variance of only 93.7.

For test 3, the mass oscillates in both horizontal and vertical directions. This effect can be observed from figure 8. Again, we examined the variance of the movements in both directions. Thus, the vertical variance equals 2083.7 and horizontal variance equals to 808.9. This result matches our observation that the magnitude of vertical displacement range is larger than that of the horizontal one.

Lastly, the mass movement trace is in test 4 is shown in figure 9. The vertical and horizontal displacement variance are 2480.8 and 680.8 respectively. One interesting observation to make is that the decaying behavior of the vertical movement. The intuition is that, the movement in and out of the vertical-horizontal plane drains energy from the system.





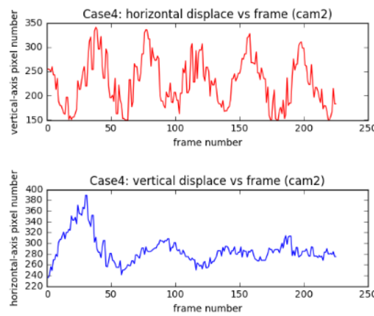



*figure 7: mass movement vs time for test 1 (top left); figure 8: mass movement vs time for test 3 (top right); figure 9: mass movement vs time for test 4 (bottom)*

## **<u>Sec. V. Summary and Conclusions</u>**

We successfully tracked all movement patterns in this experiment in horizontal and vertical axis. For each test, the tracking algorithm first analyzed the three data matrices from three cameras using standard PCA. An optimal point of view is selected based on the magnitude of the singular values of the most important modes. In our case, we picked the first two mode, which generally makes up 50-70% of the sum of singular values. Then, we cropped out the irrelevant surrounding that contains unwanted features such as people, table, chair, etc. with a rectangular scope. The size and position of the scope is selected manually by learning from the dominant principle mode that is reconstructed into a 2D grey scale image. Lastly, the actual motion tracking was achieved by finding the brightest pixel in the region of interest. This pixel is created by the flash light that is installed on the pin can, thus remain fixed relative to the mass we are interested in tracking.

One observation to make is that the results returned by SVD for all tests and all cameras have 1-2 dominant modes. This makes it easy to determine the region of interest when reconstructing them as interpretable images. However, this also leads to the limitation of the tracking model that it neglects the movement in and out of plane. The second observation is that PCA shows good promise in noisy conditions.

The valuable technical skills I obtained in processing and digesting data includes: matpotlib animation, image segmentation with skimage (various filters, line/block segmentation, water shed segmentation, etc.)