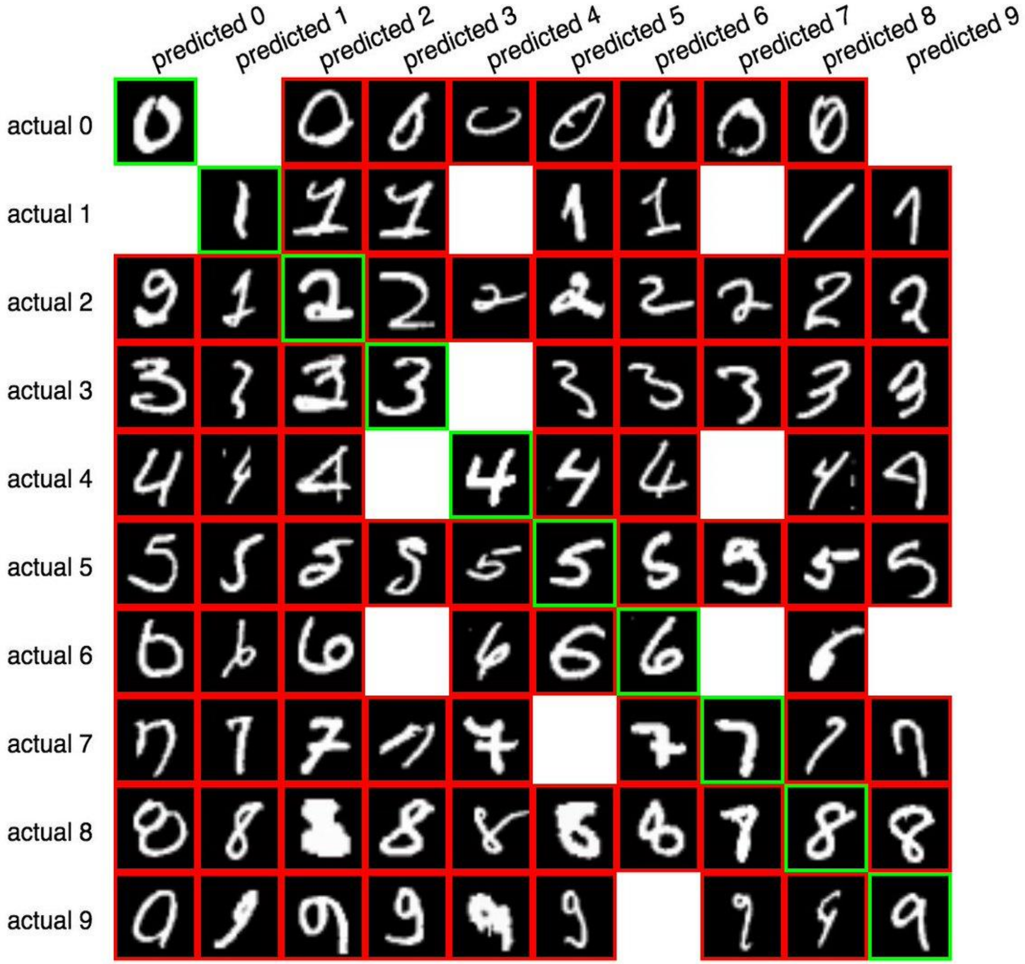
Unit V



predicted 0
predicted 1
predicted 2
predicted 3
predicted 4
predicted 5
predicted 6
predicted 7
predicted 8
predicted 9
actual 0
actual 1
actual 2
actual 3
actual 4
actual 5
actual 6
actual 7
actual 8
actual 9

- **Singular Value Decomposition**
- **Principle Component Analysis**

- **In machine learning (ML), some of the most important linear algebra concepts are the singular value decomposition (SVD) and principal component analysis (PCA).**

- **With all the raw data collected, how can we discover structures? For example, with the interest rates of the last 6 days, can we understand its composition to spot trends?**

	1 Mo	2 Mo	3 Mo	6 Mo	1 Yr	2 Yr	3 Yr	5 Yr	7 Yr	10 Yr	20 Yr	30 Yr
2/1/19	2.41	2.42	2.4	2.46	2.56	2.52	2.5	2.51	2.59	2.7	2.88	3.03
2/4/19	2.41	2.41	2.42	2.49	2.57	2.53	2.52	2.53	2.62	2.73	2.92	3.06
2/5/19	2.39	2.4	2.42	2.5	2.56	2.53	2.5	2.51	2.6	2.71	2.89	3.03
2/6/19	2.4	2.41	2.42	2.5	2.56	2.52	2.5	2.5	2.59	2.7	2.88	3.03
2/7/19	2.43	2.43	2.42	2.49	2.55	2.48	2.46	2.46	2.54	2.65	2.85	3
2/8/19	2.43	2.43	2.43	2.49	2.54	2.45	2.43	2.44	2.53	2.63	2.82	2.97

- This becomes even harder for high-dimensional raw data. It is like finding a needle in a haystack. SVD allows us to extract and untangle information.

- In simple terms, take the number 8, you can factorize this into 2x4, or 2x2x2 right?

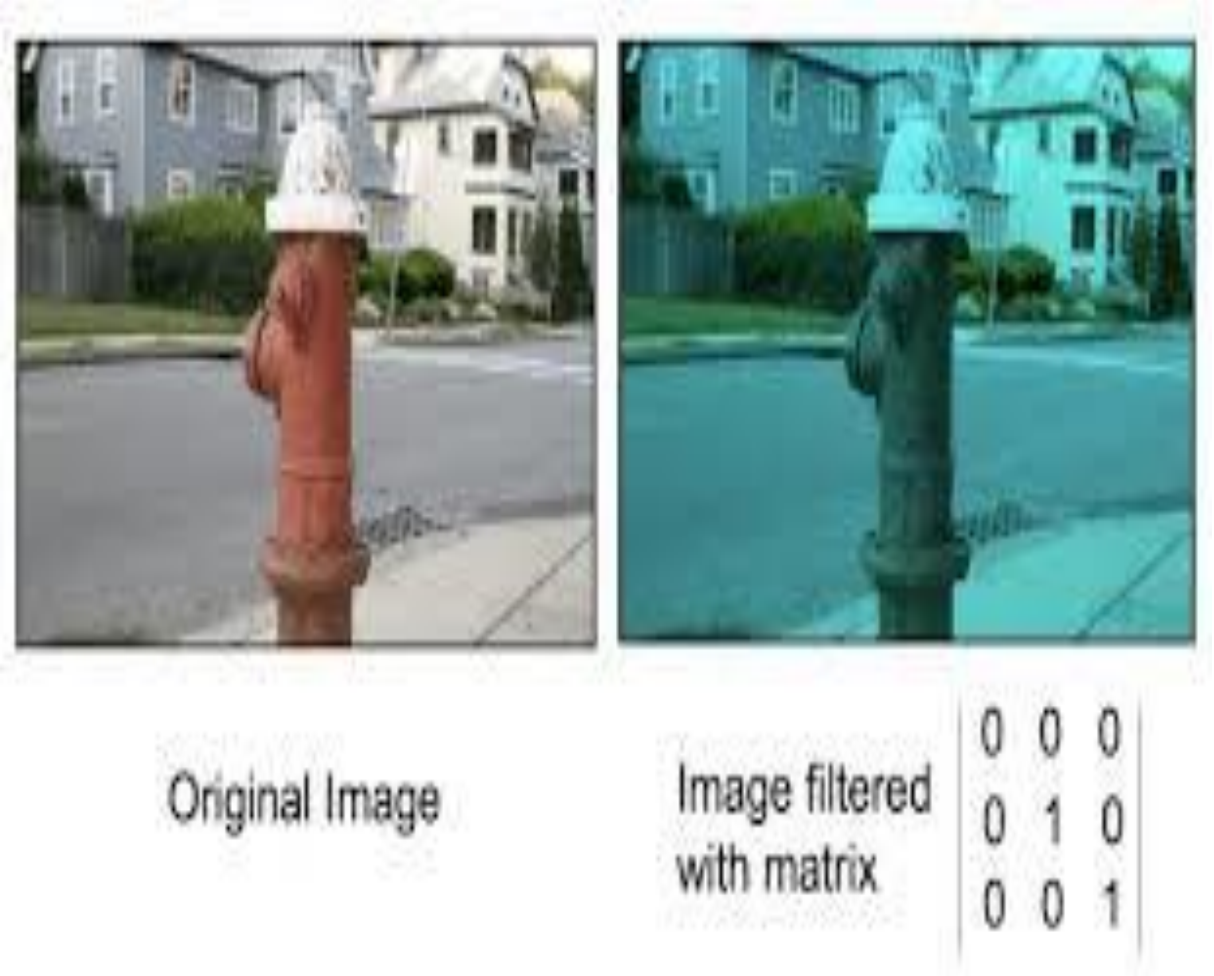
- SVD is the same thing, It is a matrix factorization technique where a matrix is decomposed into a product of a square matrix, a diagonal (possible rectangular) matrix, and another square matrix.

- **It's useful because this can actually reduce the size of the data you're dealing with. (i.e., the SVD and PCA version could be less numbers overall than the original matrix) - this can reduce RAM usage**

- **Singular Value Decomposition (SVD) and PCA is a common dimensionality reduction technique in data science**

Applications

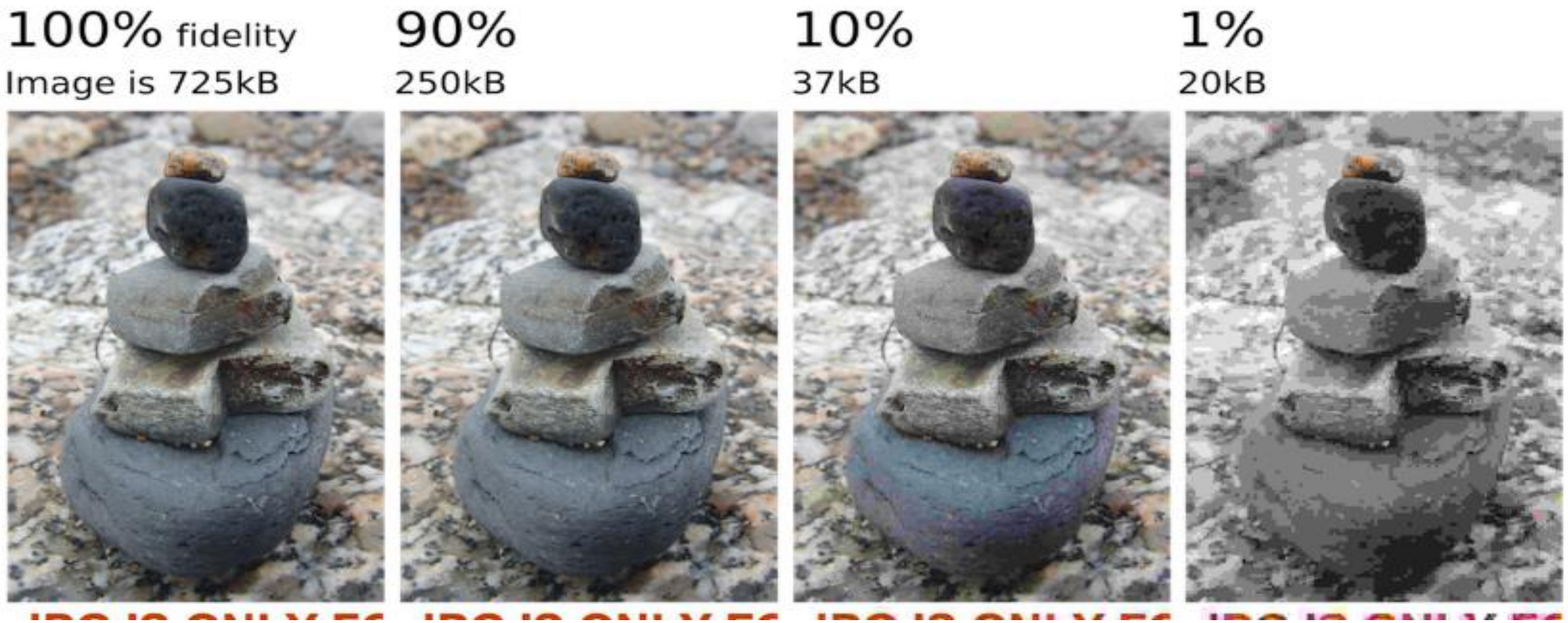
- Image Compression
- Image Recovery
- Eigenfaces
- Spectral Clustering
- Background Removal from Videos

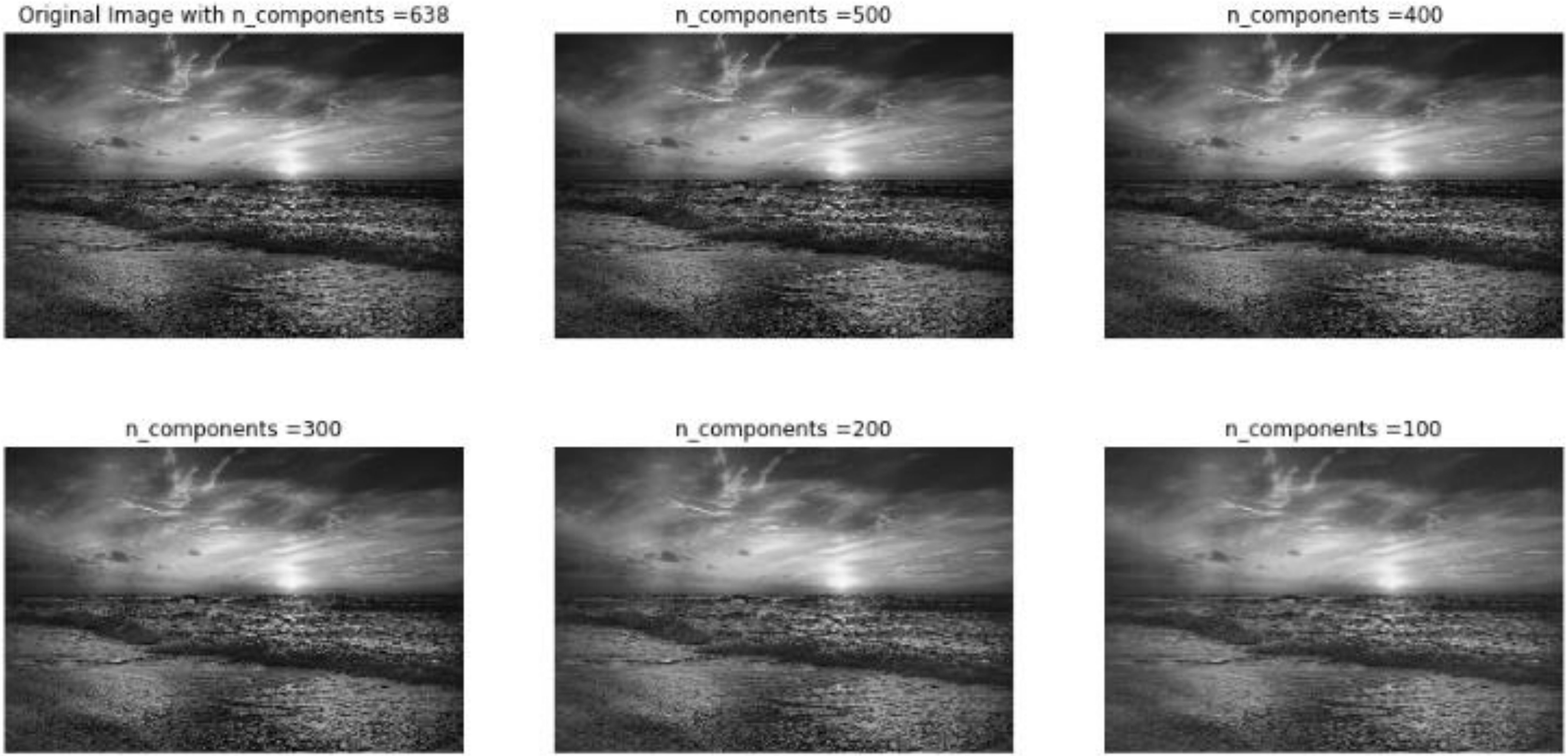


SVD for Image Compression

How many times have we faced this issue? We love clicking images with our smartphone cameras and saving random photos off the web. And then one day – no space! Image compression helps deal with that headache.

It minimizes the size of an image in bytes to an acceptable level of quality. This means that you are able to store more images in the same disk space as compared to before.




Original Image with n_components =638
n_components =500
n_components =400
n_components =300
n_components =200
n_components =100

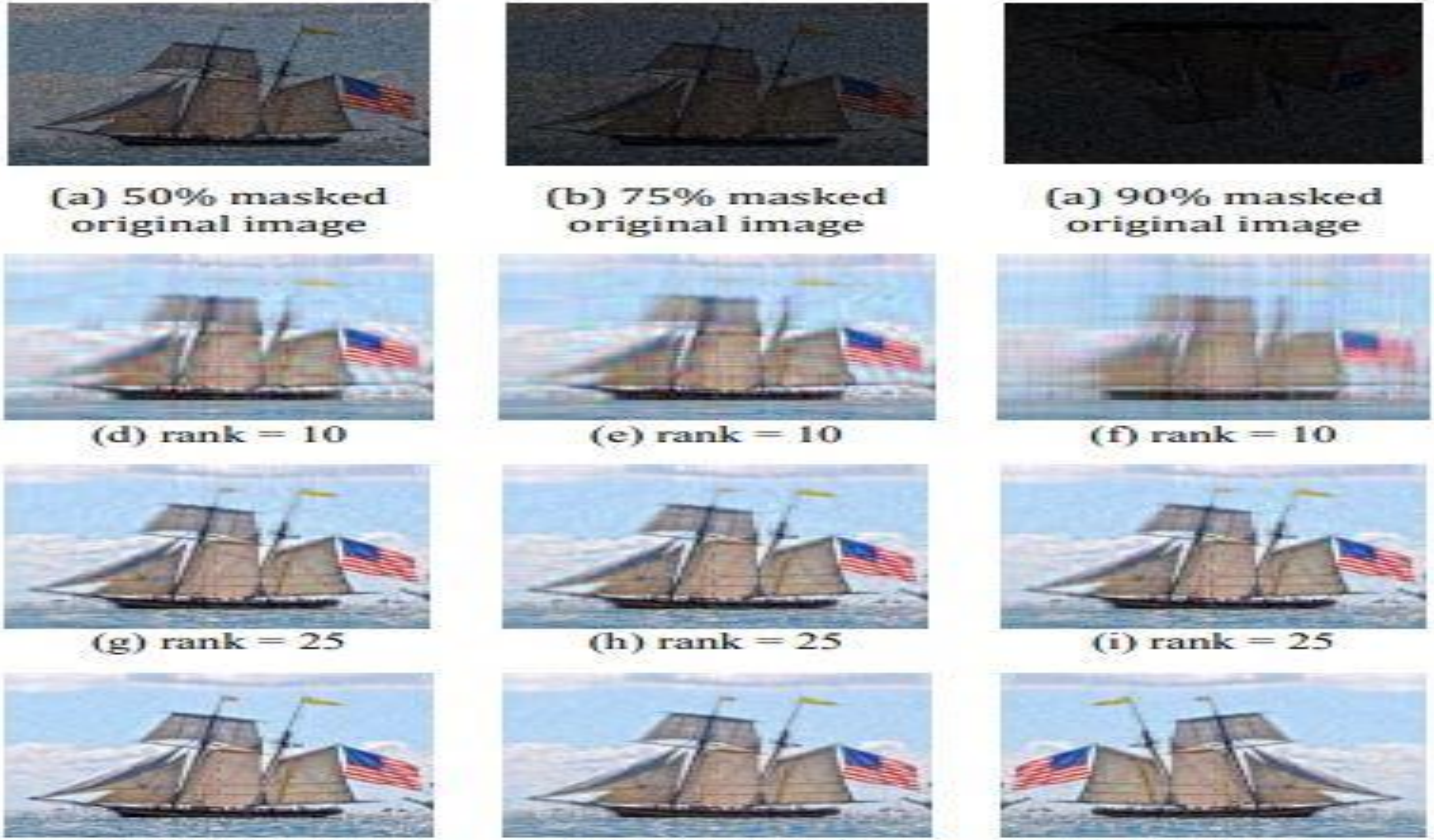
SVD for Image Recovery

- Ever clicked an image in low light? Or had an old image become corrupt? We assume that we cannot get that image back anymore. It's surely lost to the past. Well – not anymore!

- We'll understand image recovery through the concept of matrix completion

- Matrix Completion is the process of filling in the missing entries in a partially observed matrix.


A D
AD ACADÉMIE
BEFORE
AFTER



(a) 50% masked original image

(b) 75% masked original image

(a) 90% masked original image

(d) rank = 10

(e) rank = 10

(f) rank = 10

(g) rank = 25

(h) rank = 25

(i) rank = 25

(j) rank = 50

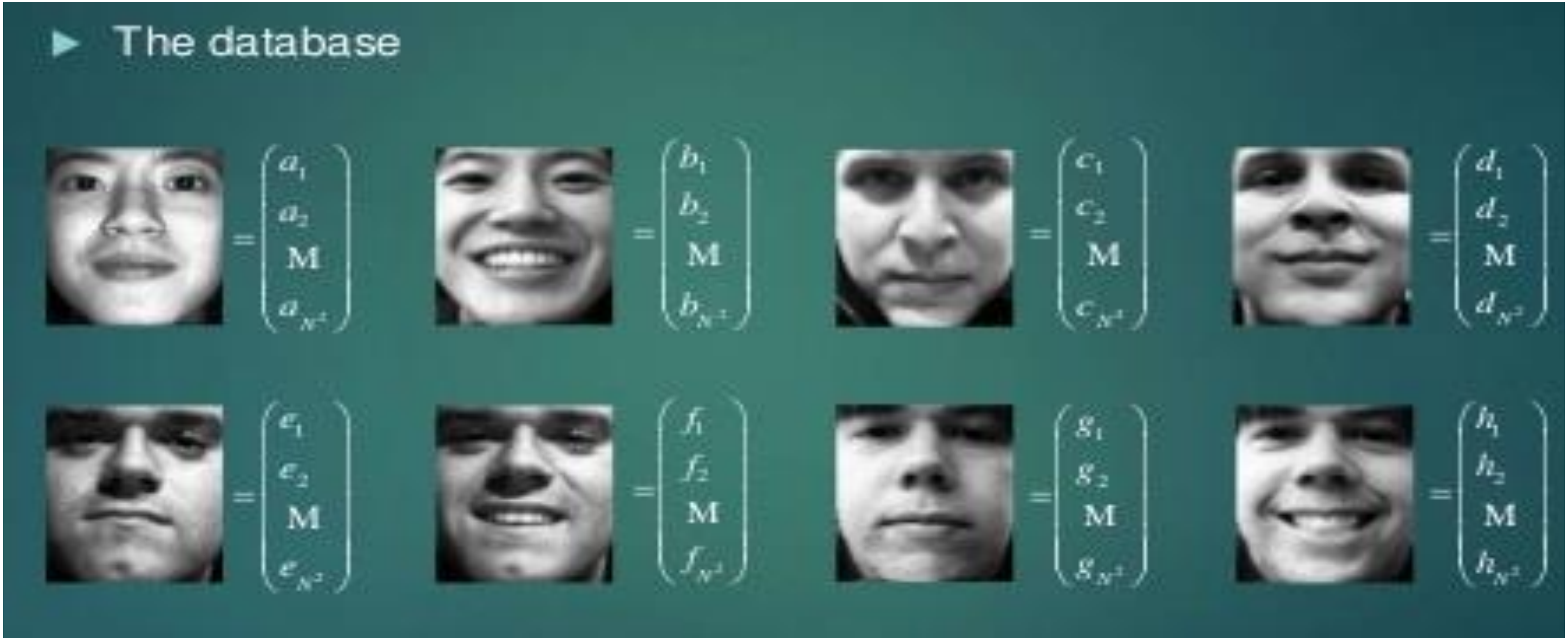
(k) rank = 50

(l) rank = 50

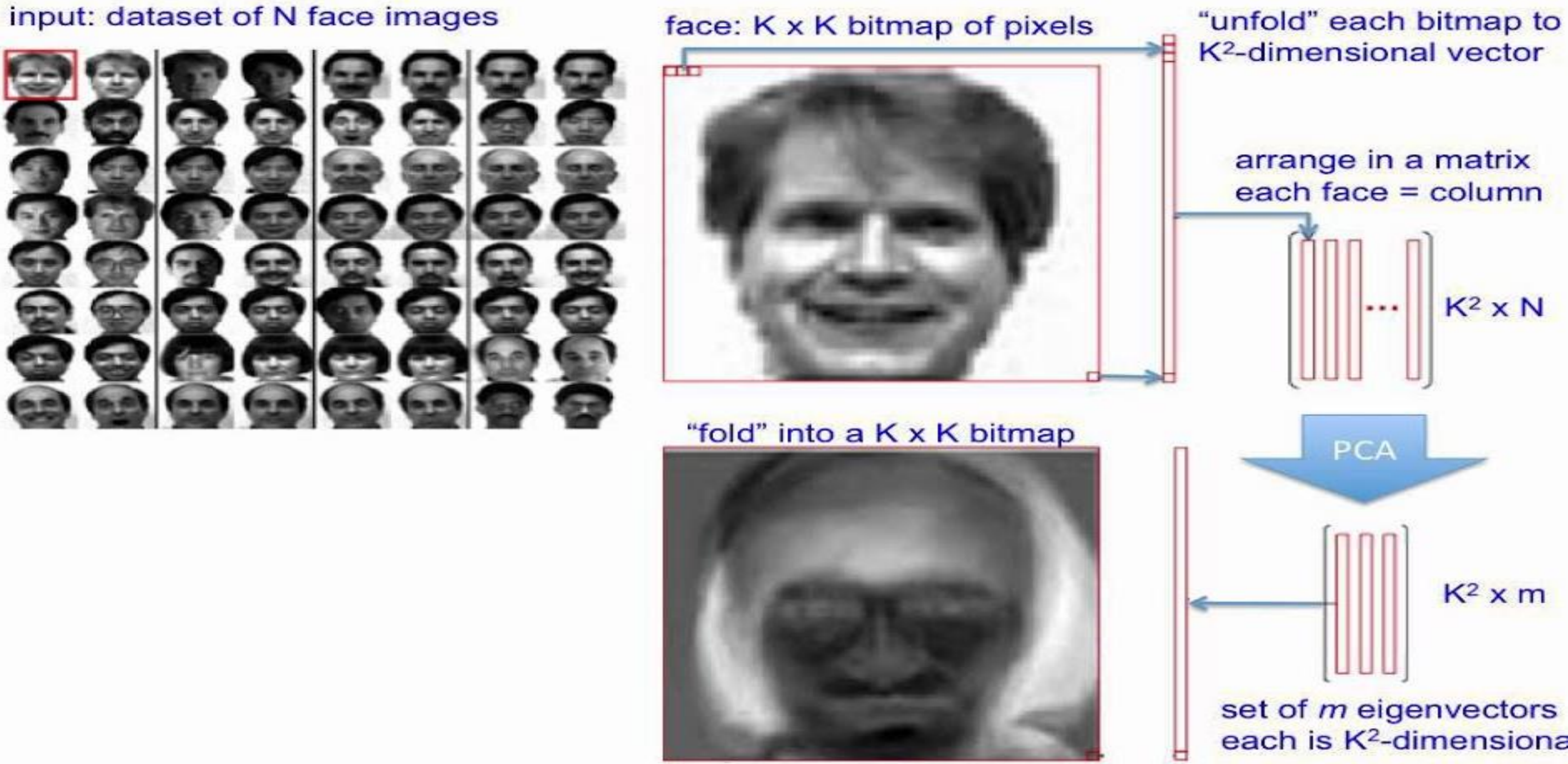
Figure 4. Recovery for a 960 × 1200 RGB image in easy, medium and hard mode

Eigenfaces

- The encoding is obtained by expressing each face as a linear combination of the selected eigenfaces in the new face space. You can find these eigenfaces using both PCA and SVD.



PCA example: Eigen Faces

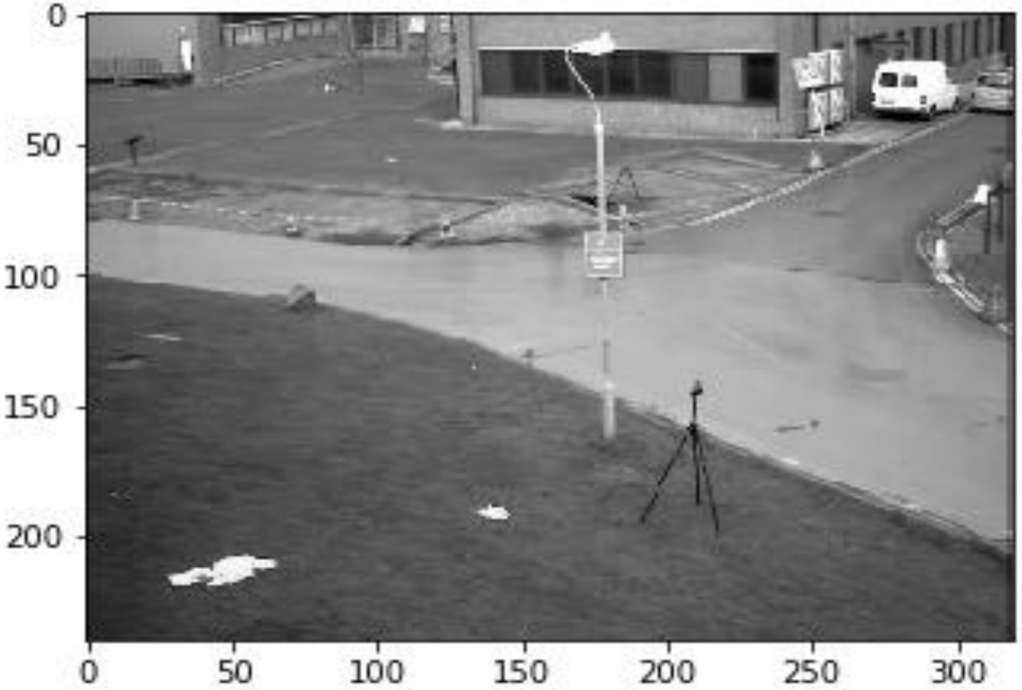


Removing Background from Videos

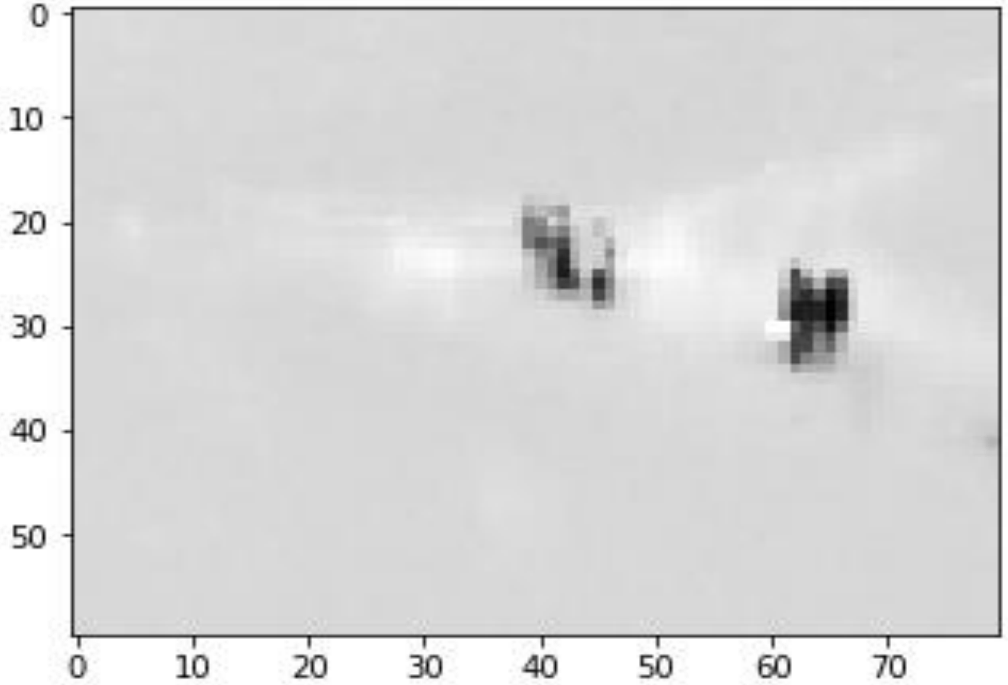
- I have always been curious how all those TV commercials and programs manage to get a cool background behind the actors. While this can be done manually, why put in that much manual effort when you have machine learning?

- Create matrix M from video – This is done by sampling image snapshots from the video at regular intervals, flattening these image matrices to arrays, and storing them as the columns of matrix M
- We can, therefore, think of M as being the sum of two matrices – one representing the background and other the foreground
- The background matrix does not see a variation in pixels and is thus redundant i.e. it does not have a lot of unique information. So, it is a low-rank matrix
- So, a low-rank approximation of M is the background matrix. We use SVD in this step
- We can obtain the foreground matrix by simply subtracting the background matrix from the matrix M

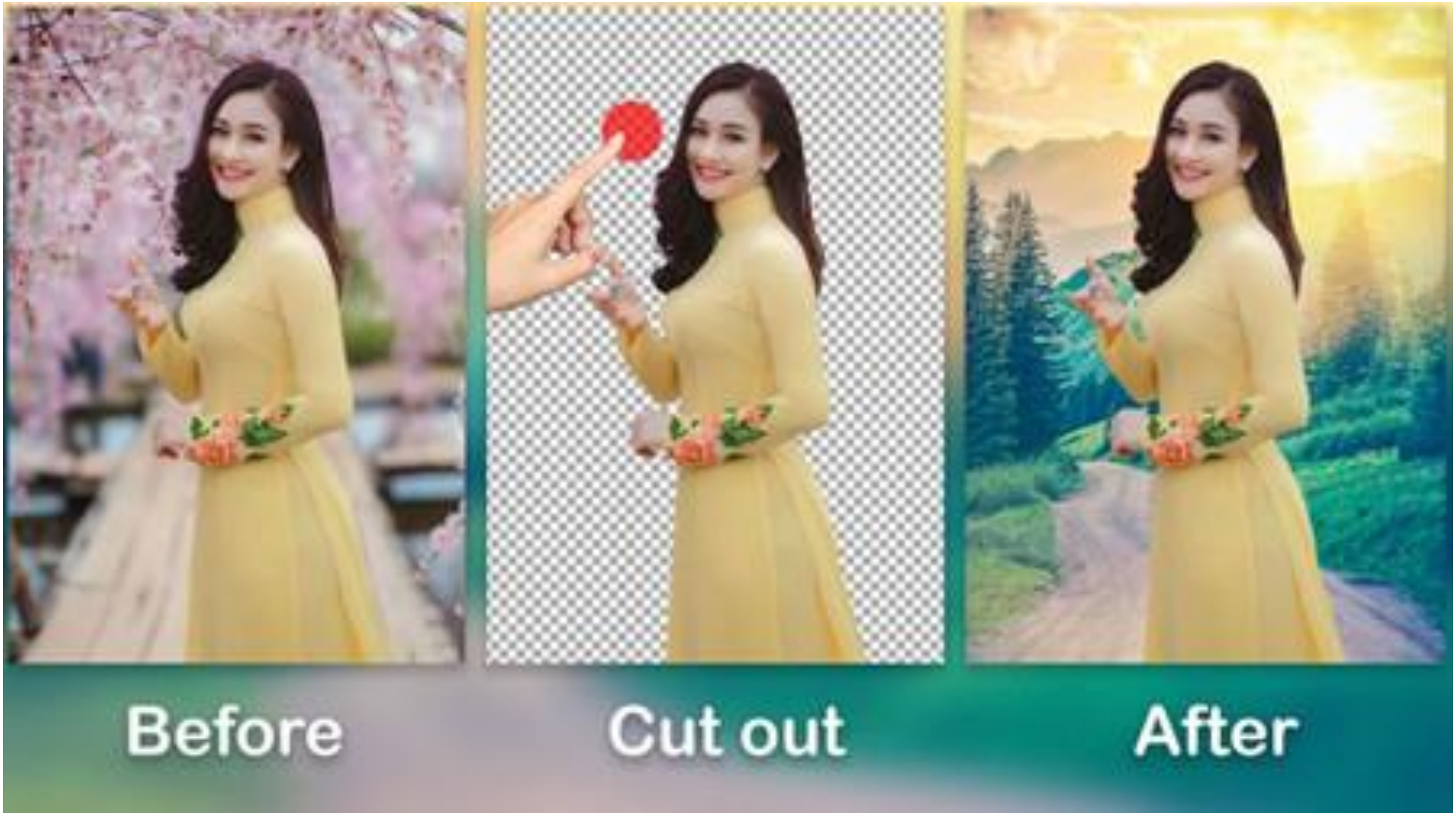
BACKGROUND REMOVAL - RANK 2 APPOXIMATION



LOW RANK BACKGROUND



FOREGROUND


Before
Cut out
After

SINGULAR VALUE DECOMPOSITION (SVD)

1. Compute $A.A^T$
2. Find out Eigen values
3. Find out Eigen Vectors
4. Find orthonormal vectors v1,v2 and v3
5. Singular values are σi=$\sqrt{\lambda_i}$
6. S=[v_1,$v_2$$v_3$]
7. Σ=diag(σi)
8. Find $u_i=\frac{1}{\sigma_i}Av_i$
9. Find U=[u_1,u_2,u_3]
10. $A=U\Sigma V^T$

Find the Singular Value Decomposition of Matrix:

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{pmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{pmatrix}$$

PRINCIPLE COMPONENT ANALYSIS (PCA)

1. Find out average of rows

2. Find out centered matrix A

3. find out Covariance Matrix S=$\frac{AA^T}{r-1}$ $(Where\ r = no.of\ rows)$

4. Find out eigen values of S.

5. Find out eigen vector of S corresponding to larger eigen value of S.

6. Normalize it.

Construct the sample covariance matrix S for the data given below:

$$\begin{pmatrix} 19 & 22 & 6 & 3 & 2 & 20 \\ 12 & 6 & 9 & 15 & 13 & 5 \end{pmatrix}$$

Find the eigenvector of S that points the most significant direction of the data

Construct the sample covariance matrix S for the data given below:

$$\begin{pmatrix} 1 & 3 & 2 & 2 \\ 2 & 4 & 5 & 1 \end{pmatrix}$$

Find the eigenvector of S that points the most significant direction of the data

Construct the sample covariance matrix S for the data given below:

$$\begin{pmatrix} 13 & 9 & 7 & 15 \\ 8 & 7 & 4 & 6 \end{pmatrix}$$

Find the eigenvector of S that points the most significant direction of the data

GOOD LUCK

FOR YOUR

EXAM AND

DO THE BEST