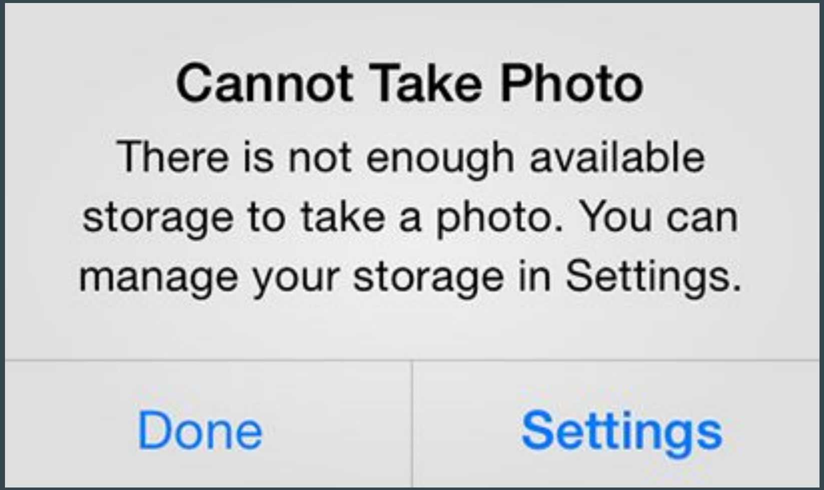
Image Compression

using Singular Value Decomposition and Principal Component Analysis

Ben Yuan & Yirui Zhu
21-344
May 13, 2019


YouTube
Facebook
Snapchat
Instagram

Whoops...


Cannot Take Photo
There is not enough available storage to take a photo. You can manage your storage in Settings.
Done
Settings

Singular Value Decomposition (SVD)

Let $A \in C^{m \times n}$ where m, n arbitrary and A not necessarily full rank, a *singular value decomposition* of A is a factorization

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

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

left singular vectors

$$U = \begin{bmatrix} u_1 & \dots & u_r & \dots & u_m \end{bmatrix}, \; \Sigma = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix}, \; V^T = \begin{bmatrix} v_1^T \\ \vdots \\ v_r^T \\ \vdots \\ v_n^T \end{bmatrix}$$

singular values

right singular vectors

How to perform SVD

Step 1.

- Multiply both sides of the equation by A^T we get

$$A^T A = V\Sigma^2 V^T$$

- Find the eigenvectors of $A^T A$

Step 2

- Find the eigenvalues of $A^T A$

- Find the square root of the eigenvalues

Step 3

- Multiply both sides of the equation by A^T we get

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

- Find the eigenvectors of AA^T

Approximation of a matrix using SVD

Given $A = U\Sigma V^T$, we can write A as a linear combination

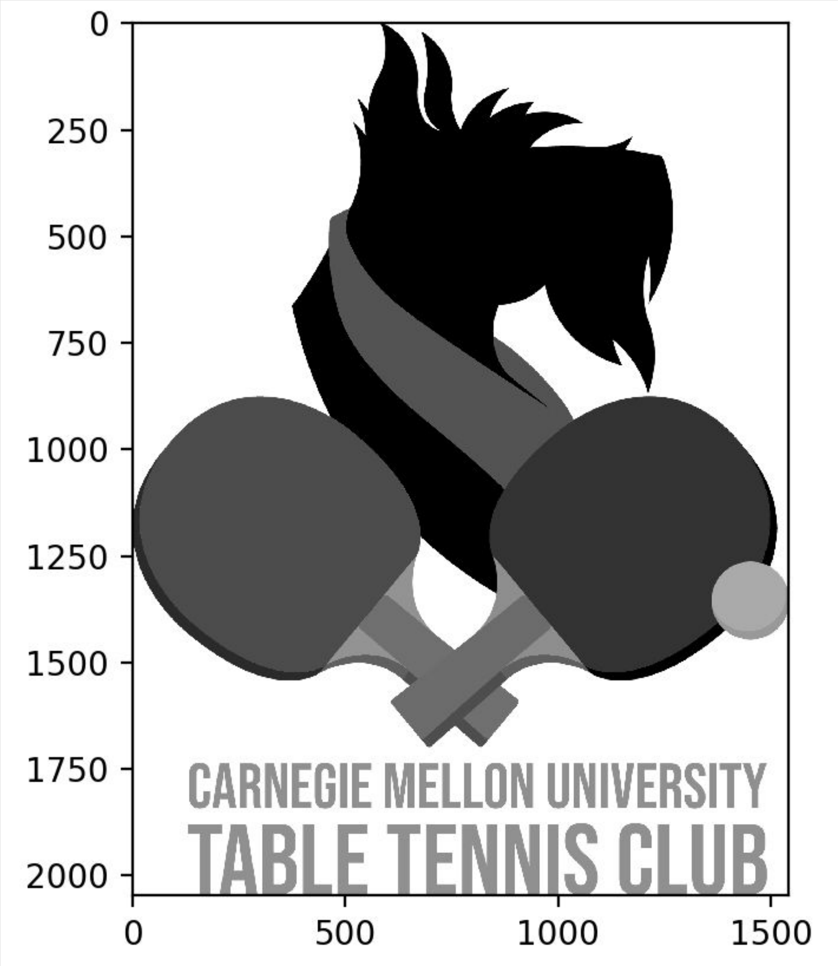
$$A = u_1\sigma_1 v_1^T + u_2\sigma_2 v_2^T + \ldots + u_i\sigma_i v_i^T + \ldots + u_n\sigma_n v_n^T$$

Application of SVD: Image Compression

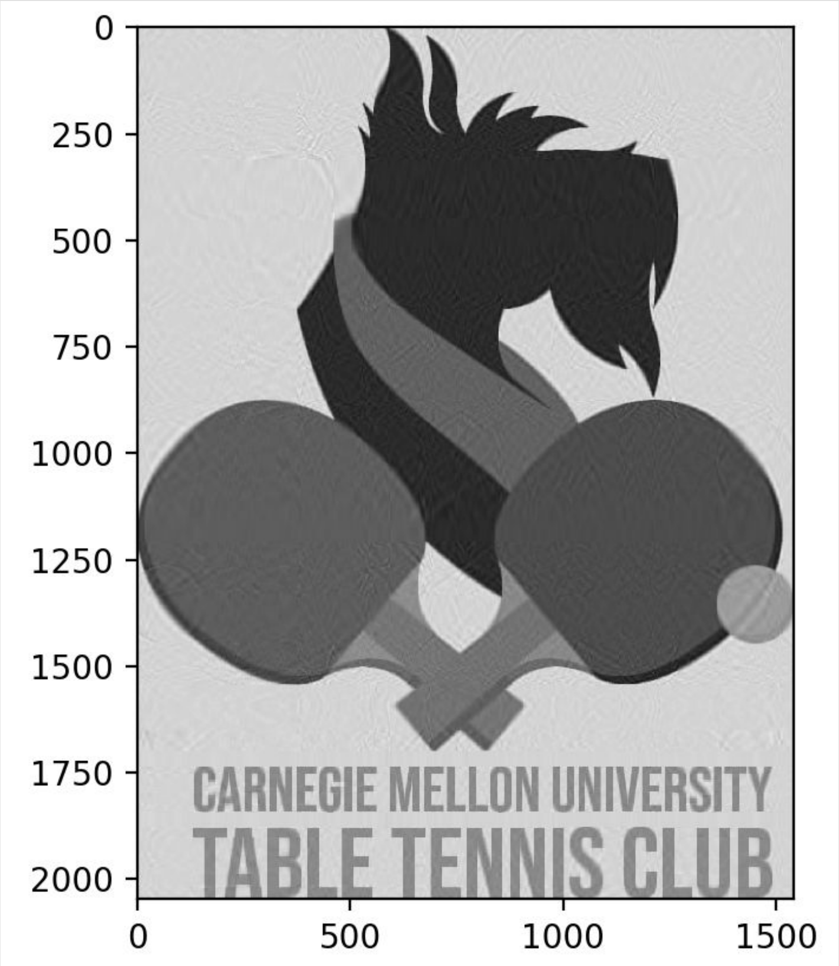
$$A = u_1\sigma_1 v_1^T + u_2\sigma_2 v_2^T + u_3\sigma_3 v_3^T + u_4\sigma_4 v_4^T + u_5\sigma_5 v_5^T$$

Application of SVD: Image Compression

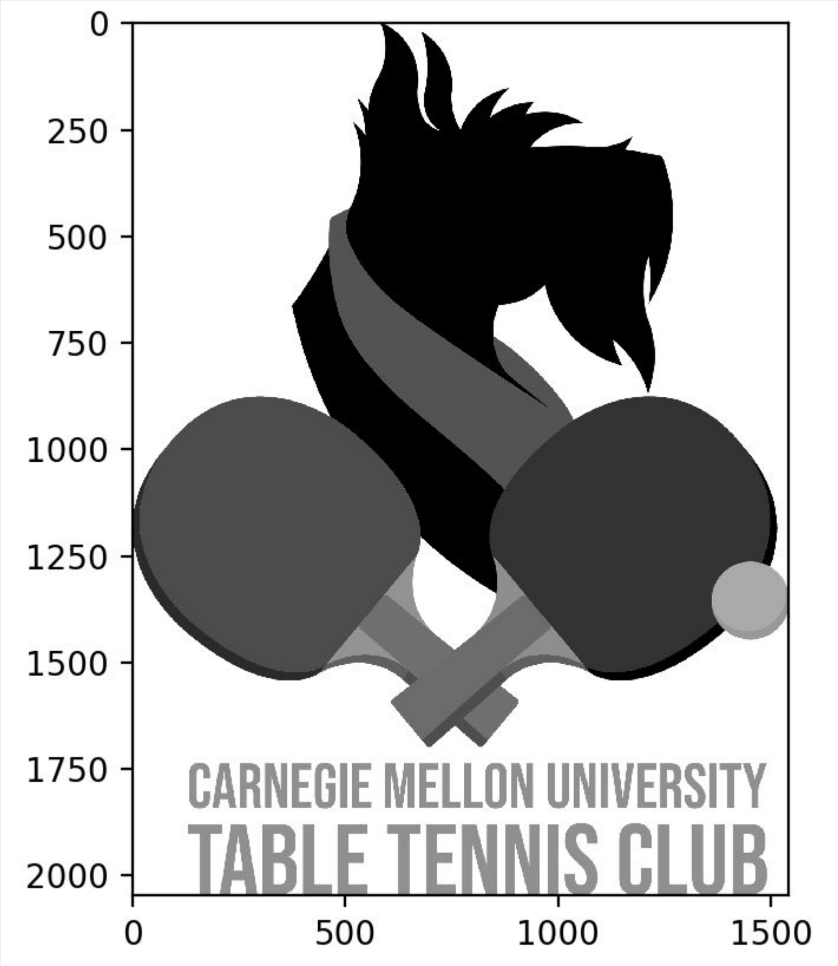
$$A = u_1\sigma_1 v_1^T + u_2\sigma_2 v_2^T + u_3\sigma_3 v_3^T + \cancel{u_4\sigma_4 v_4^T + u_5\sigma_5 v_5^T}$$



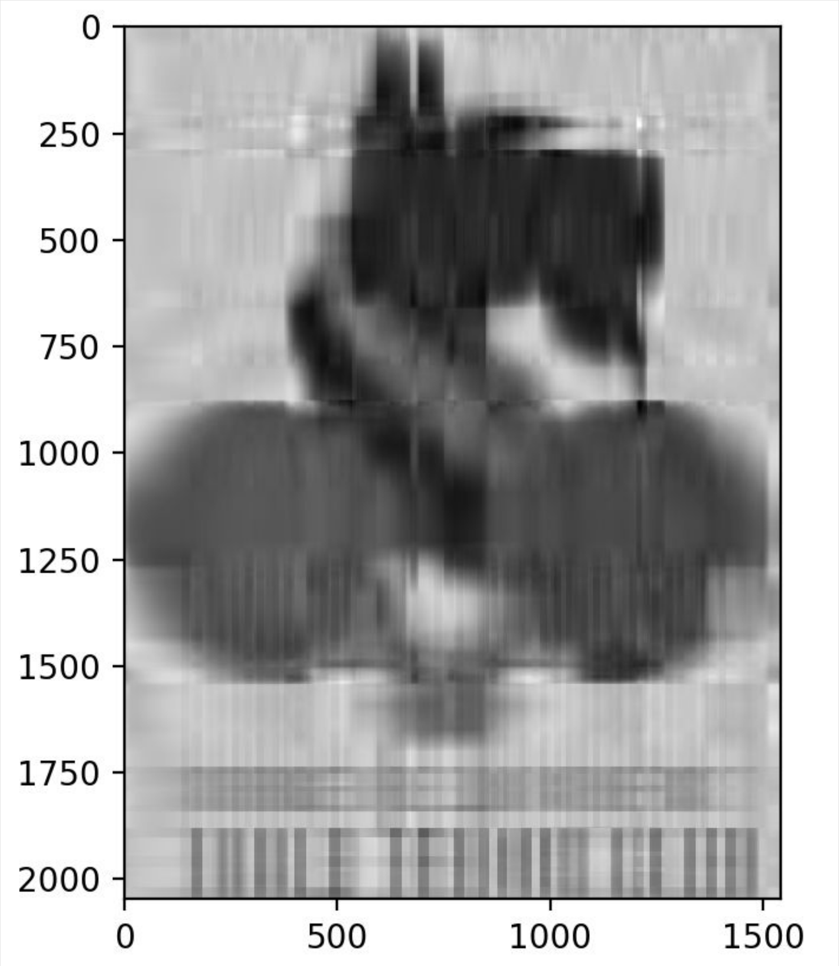
original image



n = 90

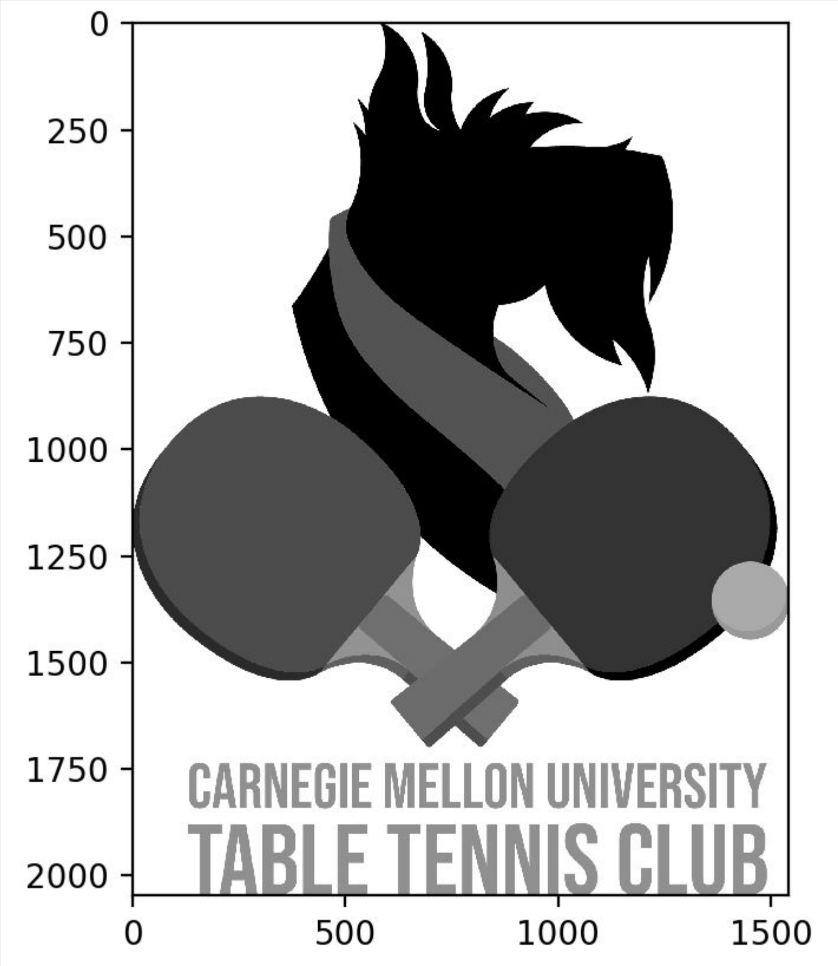


original image

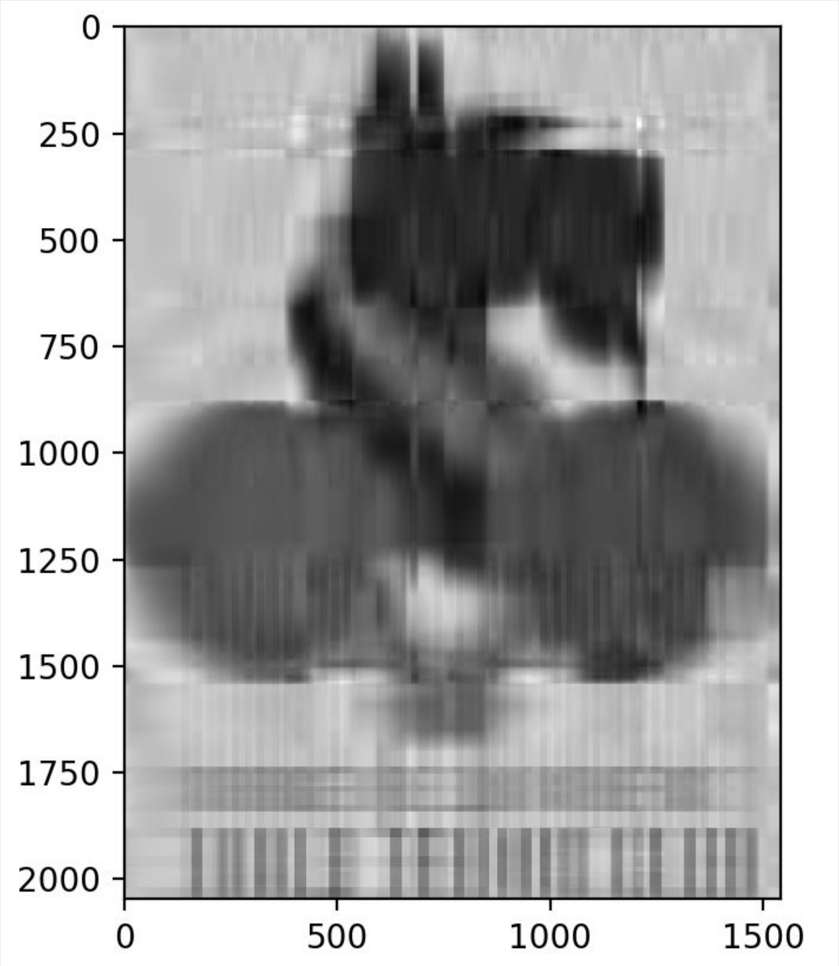


n = 10

$$A \approx \sum_{i=1}^{k} u_i \sigma_i v_i^T$$



original image



n = 10

Determine the range of k

Suppose that we want to preserve the first k terms in the linear combination, then we keep the first k rows in matrices U, Σ and V^T

$$dim(U) = m \times k, dim(\Sigma) = k \times k, dim(V^T) = k \times n$$

We want to find a k such that the total number of elements in $U\Sigma V^T$ is less than that in A

$$m \times k + k + k \times n < m \times n$$

$$\boldsymbol{k} < \frac{mn}{m+n+1}$$

Determine k with a fixed compression ratio

Let compression ratio be α, where $\alpha = \frac{m \times k + k + k \times n}{m \times n}$

With a given α, we could figure out how many terms we need to keep.

$$k = \alpha \frac{mn}{m+n+1}$$

Principal Component Analysis (PCA)

An algorithm used in machine learning and data mining to reduce high dimensional data to lower dimension data.

Ideas behind PCA

1. Dataset does not span the entire dimensional space
2. m by n matrix (m samples, n covariates)
   → n by n covariance matrix
3. Choose a subset of covariates from the dataset that explain as much of the variation in the data as possible (principal components)

Covariance Matrix

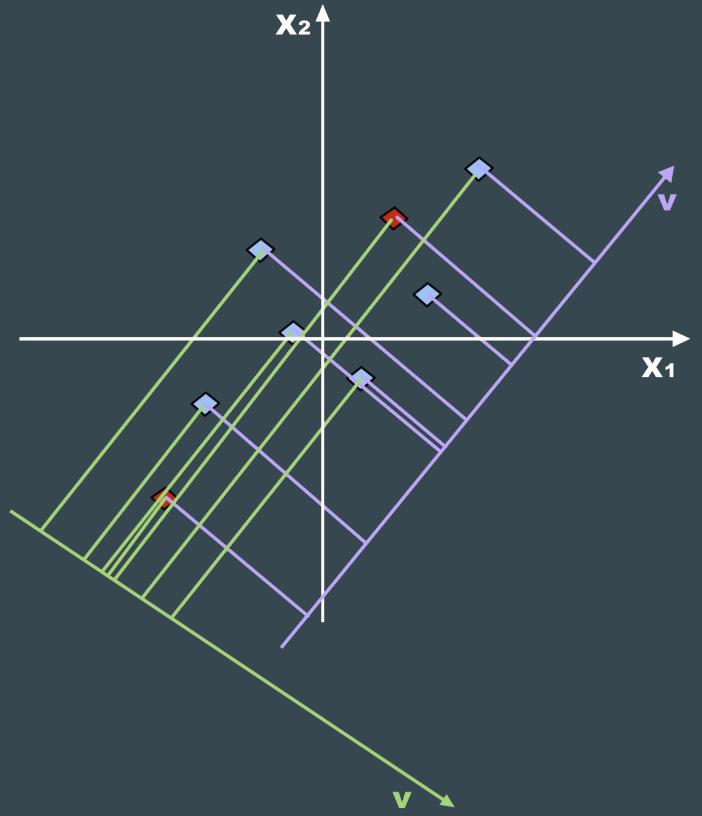
$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{n-1}$$

e.g. covariance matrix of a dataset with 3 covariates (x, y, z)

$$\begin{bmatrix} cov(x,x) & cov(x,y) & cov(x,z) \\ cov(y,x) & cov(y,y) & cov(y,z) \\ cov(z,x) & cov(z,y) & cov(z,z) \end{bmatrix}$$

Why More Variability?

- We DON'T WANT cases where two points are far away in 2d-space, but close to each other in 1d-space
- Pick v to maximize variability → minimizes number of undesirable cases



Finding Principal Components

Compute eigenvalues of covariance matrix

→ eigenvalue with the largest absolute value indicates that data has the largest variance along its eigenvector

Algorithm

1. For each column x_i, set $x_i = x_i - \bar{x}$
2. Compute the covariance matrix C of adjusted dataset matrix
3. Compute eigenvalues and eigenvectors of C

Eigenvalues and Explained Variance

Proportion of Explained Variance along the k-th eigenvector

$$\frac{\lambda_k}{\sum_{i=1}^{n} \lambda_i} \tag{1}$$

Proportion of Explained Variance along the first k eigenvectors

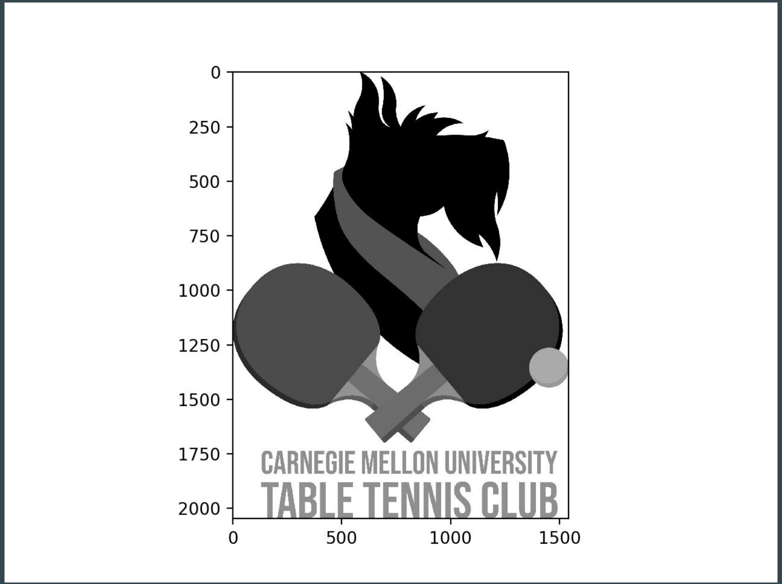
$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \tag{2}$$

Determine k

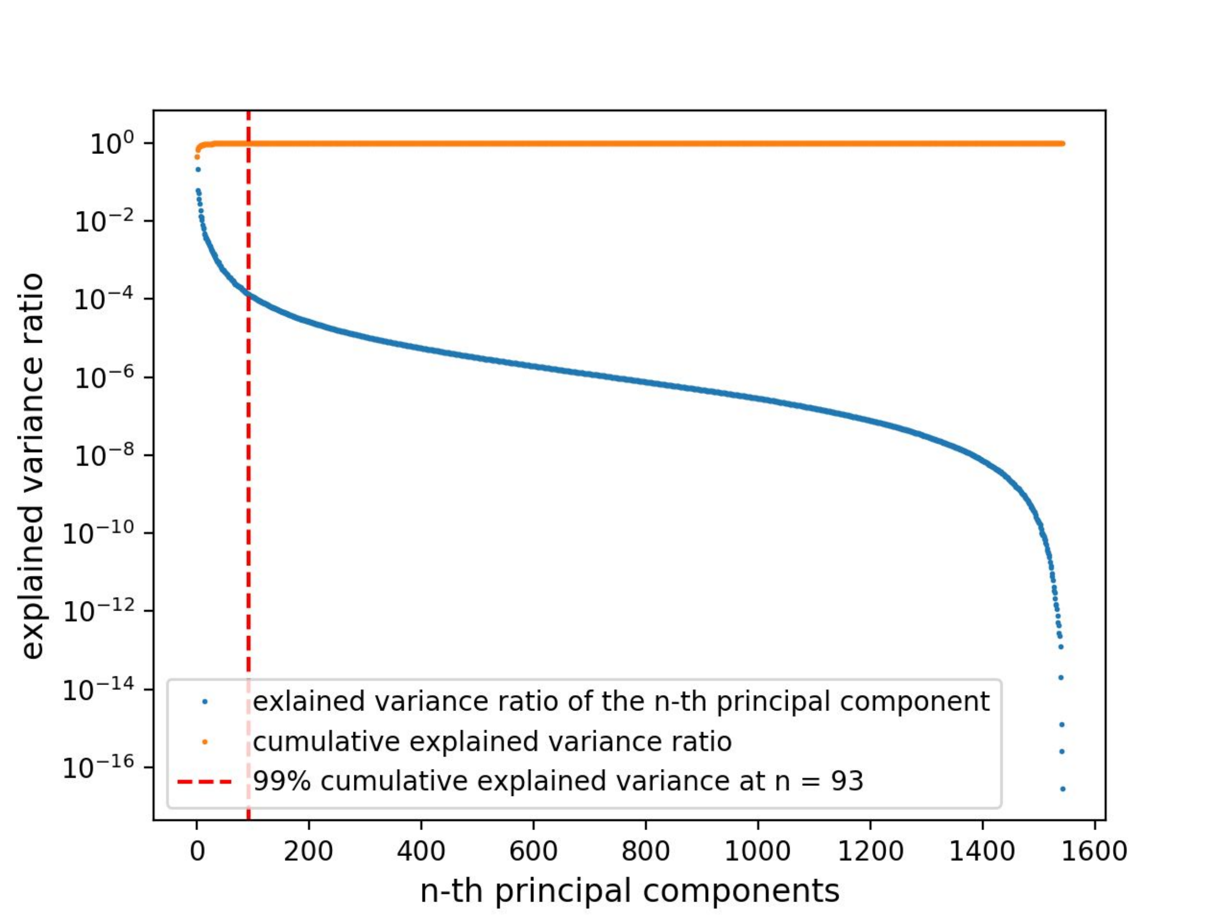
Empirically, we look for the minimum k such that

$$\frac{\sum_{i=1}^{k} \lambda_k}{\sum_{i=1}^{n} \lambda_i} \geq 0.99$$

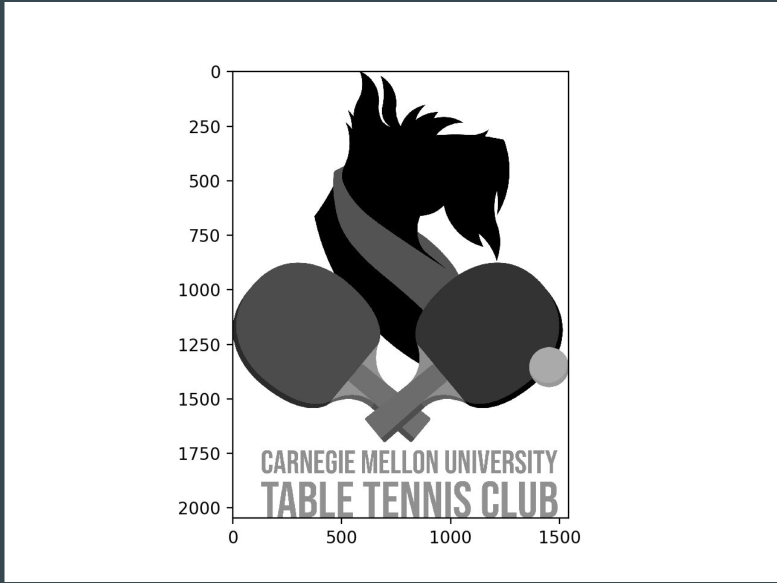
Example



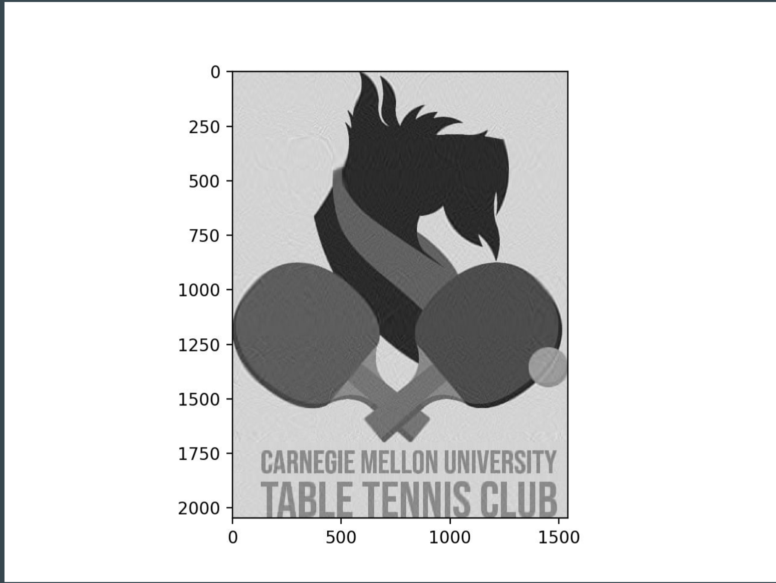
original


explained variance ratio
n-th principal components
exlained variance ratio of the n-th principal component
cumulative explained variance ratio
99% cumulative explained variance at n = 93

Result

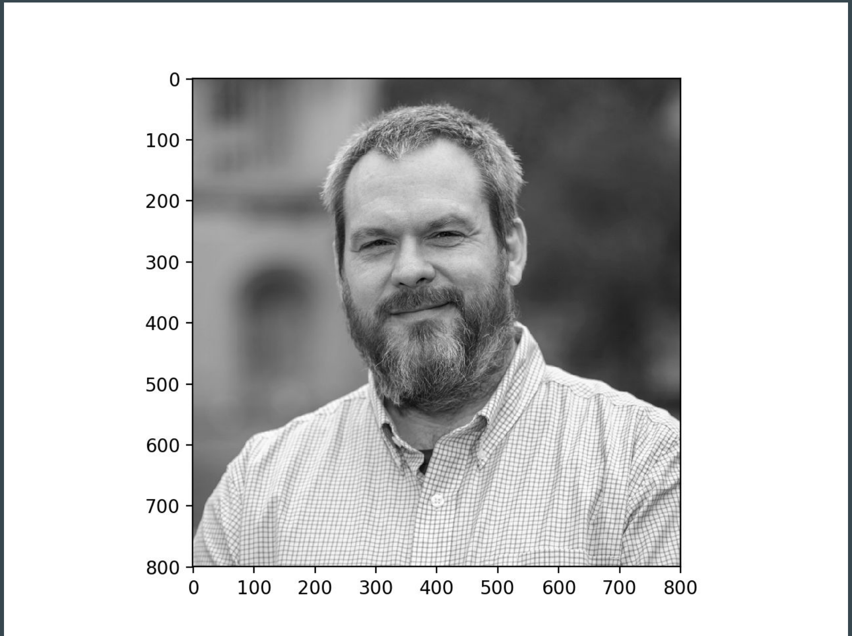


original

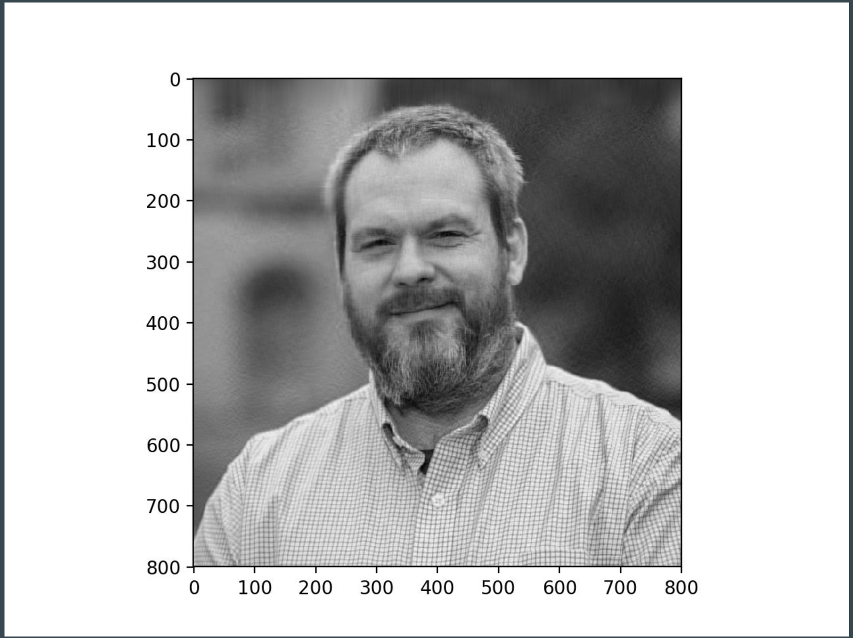


$k = 93$

One More Example



original

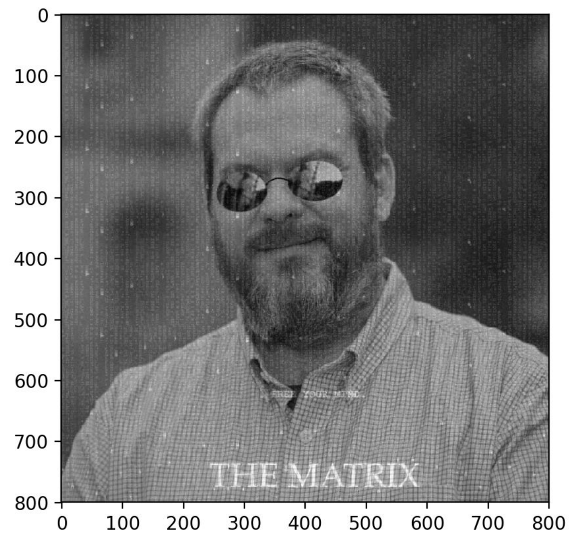


n = 100

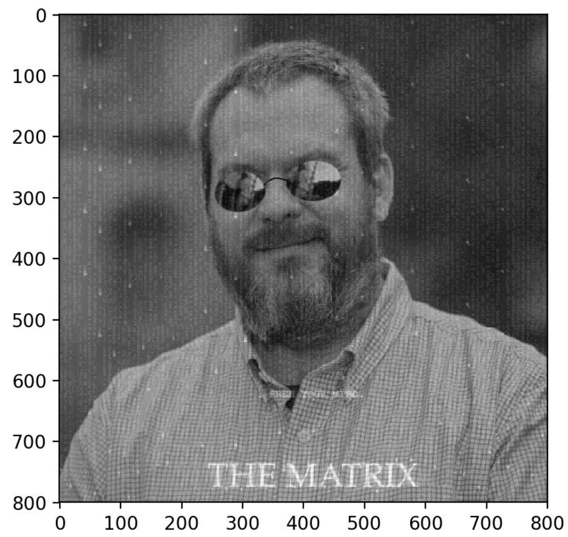

THE
MATRIX


FREE YOUR MIND.
THE MATRIX

Last Example



original



n = 170

Future work

1. Apply to Colored Images
2. Measure on Quality of Compressed Images
3. Interpretability
4. How to Set Proportion of Explained Variance

References

SVD:

- https://www.cmi.ac.in/~ksutar/NLA2013/imagecompression.pdf (principles)
- https://www.frankcleary.com/svdimage/ (code)

PCA:

- http://deeplearning.stanford.edu/wiki/index.php/%E4%B8%BB%E6%88%90%E5%88%86%E5%88%86%E6%9E%90 (principles)
- https://blog.csdn.net/u012162613/article/details/42177327 (code)
- https://www.youtube.com/watch?v=6Pv2txQVhxA (visualization)