Designing Compression Methods Using Singular Value Decomposition And Low Rank Matrix Approximation

Jonathan J. Allarassem
Grove City College
Grove City, PA, USA
allarassemjj20@gcc.edu

November 14, 2025

Abstract

This paper delves into the utilization of linear algebra as a foundation for crafting compression methods. The study investigates the application of linear algebra concepts for image compression specifically. This paper outlines how this can be done using Singular Value Decomposition and Low-Rank Matrix Approximation.

1 Introduction

The success of numerous tech companies hinges on their capacity to seamlessly connect individuals with personalized products or content. Profits amounting to millions have been generated by firms that have mastered the creation of such systems. A crucial initial step in this endeavor frequently revolves around large-scale data compression. This paper aims to delve into how insights from Linear Algebra can facilitate such compression outcomes. Following this introduction, the paper will explore the broad spectrum of applications, subsequently delving into the theoretical underpinnings in Linear Algebra. Finally, it will showcase select results and explore intriguing concepts.

2 General Applications

Singular Value Decomposition (SVD) is a powerful matrix factorization technique with numerous applications across various fields. SVD can be used for reducing the dimensionality of data while preserving its essential structure [2]. SVD can also be applied to compress images by representing them in terms of a smaller number of singular vectors and values, thus reducing storage space while retaining important visual information [1]. Other examples include noise reduction. SVD is utilized for denoising signals or images by separating noise from the underlying signal structure. This is particularly useful in image processing and signal processing applications.

3 Theoretical Foundations

This methods arises as a consequence of two main theorems in Linear Algebra.

3.1 Definitions

Before looking at these two theorems, let us define a couple of important terms (See Pearson Notebook for References).

Definition 1.1: For a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, consider $\boldsymbol{\Gamma} = \boldsymbol{A}^T\boldsymbol{A}$. We can see that $\boldsymbol{\Gamma} \in \mathbb{R}^{n\times n}$ and

$\mathbf{\Gamma}$ is symmetrical so there exists $\{\lambda_i\}_{i\in\mathbb{N}}$ the set of eigenvalues for $\mathbf{\Gamma}$ such that $\forall\lambda_i,\ \lambda_i \geq 0$. Let $\sigma_i = \sqrt{\lambda_i}$ so that $\forall\sigma_i, \sigma_i \geq 0$. The set $\{\sigma_i\}_{i\in\mathbb{N}}$ contains the **singular values** of the matrix $\boldsymbol{A}$.

Definition 1.2: A matrix $\boldsymbol{A}$ is of rank k if the largest linearly independent subset of columns of $\boldsymbol{A}$ has size k. That is, all n columns of $\boldsymbol{A}$ arise as linear combinations of only k of them. Which is equivalent to saying that k is the size of the largest linearly independent subset of rows in $\boldsymbol{A}$.
In other words, a matrix $\boldsymbol{A}$ has rank k if it can be written as the sum of k rank-one matrices and cannot be written as the sum of $k-1$ or fewer rank-one matrices.

Definition 1.3: For a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, the **Frobenius norm** is the norm of the matrix $\mathbf{A}$ computed in the following way:

$$\|\mathbf{A}\|_F = \sqrt{\sum_i^m \sum_j^n \|a_{ij}\|^2} = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2(\mathbf{A})} \tag{1}$$

Where $\|a_{ij}\|$ is the absolute value of a_{ij} and $\sigma_i(\boldsymbol{A})$ are the singular values of $\boldsymbol{A}$.

3.2 Theorems

This part presents the theorems that will be using to achieve the task. (**See [1, 2] for reference**)
Theorem 1.1: For a matrix $\boldsymbol{R} \in \mathbb{R}^{m\times n}$, there exists two orthogonal matrices $\boldsymbol{U} \in \mathbb{R}^{n\times n}$ and $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ and a non-negative, "diagonal" matrix $\mathbf{\Sigma} \in \mathbb{R}^{n\times d}$ (of the same size as R) such that

$$\boldsymbol{R}_{m\times n} = \boldsymbol{U}_{n\times n}\mathbf{\Sigma}_{n\times d}\boldsymbol{V}^T_{d\times d} \tag{2}$$

Remark: This is called the **Singular Value Decomposition** of $\boldsymbol{R}$

1. The diagonals of $\mathbf{\Sigma}$ are called the **singular values** of $\boldsymbol{R}$ (often sorted in decreasing order).
2. The columns of $\boldsymbol{U}$ are called the **left singular values** of $\boldsymbol{R}$.
3. The columns of $\boldsymbol{V}$ are called the **right singular values** of $\boldsymbol{R}$.

Theorem 1.2: According to **Definition1.2** for a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ if $r = Rank(\mathbf{A})$ then

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \tag{3}$$

Hence if we consider $k \leq r$ then

$$\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T \tag{4}$$

$\boldsymbol{A}_k$ is the best approximation for $\boldsymbol{A}$ of rank k:

$$\|\boldsymbol{A} - \boldsymbol{A}_k\| \leq \|\boldsymbol{A} - \boldsymbol{B}\| \quad \forall \boldsymbol{B} \in \boldsymbol{Rank}^{(k)} \tag{5}$$

Where $\boldsymbol{Rank}^{(k)}$ is the set of all matrices of rank k. And the error between the matrices is a known value:

$$\|\boldsymbol{A} - \boldsymbol{A}_k\|_F = \sqrt{\sum_{k+1}^{r} \sigma_i^2} \tag{6}$$

Theorem 1.3: This theorem is applying **Theorem 1.2** on matrix factorized using **Theorem 1.1**. According to (2), for a given matrix $\boldsymbol{R} \in \mathbb{R}^{m \times n}$, there exists two orthogonal matrices $\boldsymbol{U} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{V} \in \mathbb{R}^{d \times d}$ and a non-negative, "diagonal" matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times d}$ (of the same size as R) such that:

$$\boldsymbol{R} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{7}$$

Let us define $\boldsymbol{R}_k$, $\boldsymbol{U}_k$, $\boldsymbol{\Sigma}_k$ and $\boldsymbol{V}_k$ as the first k columns of the respective matrices such that:

$$\boldsymbol{R}_k = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^T \tag{8}$$

According to **Theorem 1.2** we know that the best approximation to $\boldsymbol{R}$ of rank k is $\boldsymbol{R}_k$ which is the first k columns of the matrix $\mathbf{R}$. One advantage with this method for reducing the number of dimensions of $\mathbf{A}$ is that it allows us to control the error through the formula at (6).

3.3 RGB Image Compression

3.3.1 Formalizing Image Compression

An RGB image can be stored as a set of 3 matrices of the same size. Assume $\mathbf{I}$ is the matrix representing an image and $\boldsymbol{G}$, $\boldsymbol{B}$, $\boldsymbol{R}$ three matrices holding the red, blue and green values of each pixel, such that ($\boldsymbol{G}$, $\boldsymbol{B}$, $\boldsymbol{R}$) is a pixel value. According to **Theorem 1.1**, we can rewrite each of these matrices as the product of 3 other matrices:

$$\boldsymbol{G} = \boldsymbol{U}^{(g)}\boldsymbol{\Sigma}^{(g)}(\boldsymbol{V}^{(g)})^T \tag{9}$$

In the same way we can decomposed $\boldsymbol{R}$ and $\boldsymbol{B}$ but for sake of ease, let use simply consider $\mathbf{I}_g$ and apply the same ideas to the rest of the matrices. According to **Theorem 1.2**, we can find the best k-ranked estimation of the Red, Green and Blue matrices. And according to that same theorem we can tune how high k needs to be. The error can amount to the distance between $\boldsymbol{G}$ and $\boldsymbol{G}_k$

$$\|\boldsymbol{G} - \boldsymbol{G}_k\|_F = \sqrt{\sum_{k+1}^{r} \sigma_i^2} \tag{10}$$

It is easy to see how if we knew the maximum error $\hat{\epsilon}$ allowed for our estimation then we could infer $\hat{k}$ the best rank value in the following way:

$$\hat{k} = \underset{k \in \mathbb{N}}{\arg\min} \|\hat{\epsilon} - \phi(k)\| \tag{11}$$

With $\phi(k)$ a function taking in the rank value and normalizing the error rate using the $\boldsymbol{\Sigma}$ diagonal matrix.

3.3.2 Some results and interesting ideas

Below is a display of a picture for different values of k. We can see that even if the quality of the picture slightly decreases when we reduce the k value, we need to get it down by a factor of at least 20 for the image to start becoming seriously blurry.
Something interesting about this approach is that, it does not just apply to images. If the data used can be represented as a matrix, then we can always compress it to a lower rank matrix and use that estimation instead.

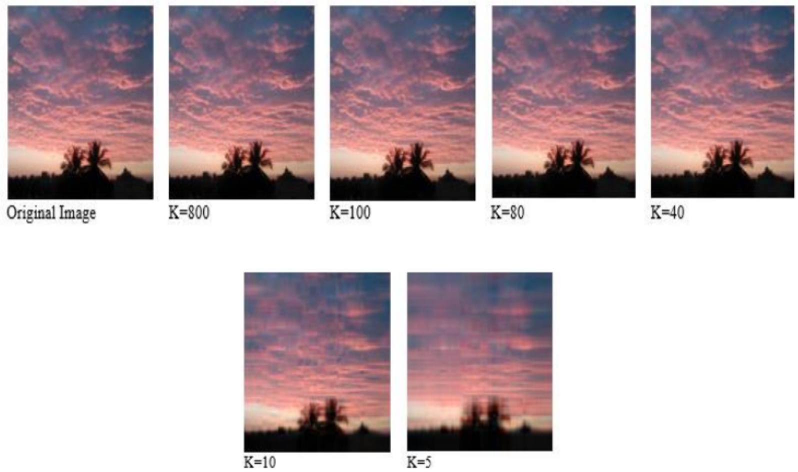

Figure 1: Quality of the image in terms of k value [3]

4 References

1. H Prasantha, H Shashidhara, M Balasubramanya, *Image Compression Using SVD*, International Conference on Computational Intelligence and Multimedia Applications, **2008**, 143, DOI 10.1109/ICCIMA.2007.386

2. F Colacea, D Conteb, M De Santoa, M Lombardia,D Valentino, *A content-based recommendation approach based on singular value decomposition*, Connection Science, **2022**, 2158–2176, DOI 1080/09540091.2022.2106943

3. H Swathi, S Sohini, G Gopichand, *Image compression using singular value decomposition*, IOP Conf. Series: Materials Science and Engineering, **2017**, 263, DOI 10.1088/1757-899X/263/4/042082