Singular Value Decomposition (SVD) Based Image Compression

AI25BTECH11006 - Bonala Nikhila

Summary of Strang's Video

In Gilbert Strang's MIT lecture on Singular Value Decomposition (SVD), every real matrix A can be decomposed as:

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

where U and V are orthogonal matrices, and Σ contains the singular values.

- Columns of U are eigenvectors of AA^T
- Columns of V are eigenvectors of A^TA
- Singular values $\sigma_i = \sqrt{\lambda_i}$, where λ_i are eigenvalues of A^TA or AA^T

This decomposition reveals the geometry of linear transformations. Large singular values correspond to directions of maximum stretch. In image compression, using the top k singular values preserves the most important information while discarding small details.

Algorithm

Given a grayscale image matrix A of size $m \times n$:

1) Compute A^T and form A^TA, a symmetric positive semi-definite matrix.
2) Compute eigenvalues λ_i and eigenvectors v_i of A^TA.
3) Singular values: $\sigma_i = \sqrt{\lambda_i}$
4) Left singular vectors: $u_i = \frac{1}{\sigma_i}Av_i$
5) Construct truncated SVD using top k singular values:

$$A_k = U_k\Sigma_k V_k^T$$

Pseudocode

Read grayscale image as matrix A
Compute $M = A^T A$
for each $i = 1 \ldots k$ **do**
    Find largest eigenvalue λ_i of M using Power Iteration
    Compute corresponding eigenvector v_i
    Compute left singular vector $u_i = \frac{1}{\sqrt{\lambda_i}} A v_i$
    Deflate $M := M - \lambda_i v_i v_i^T$
end for
Form $A_k = U_k \Sigma_k V_k^T$
Reconstruct image from A_k

Algorithm Comparison and Choice

- Full SVD: Computes all singular values, but computationally expensive ($O(mn^2)$).
- Power Iteration: Efficient for computing top k singular values.
- Chosen algorithm: Truncated SVD with Power Iteration for efficiency and memory savings while retaining important image features.

1 Reconstructed Images for Different k

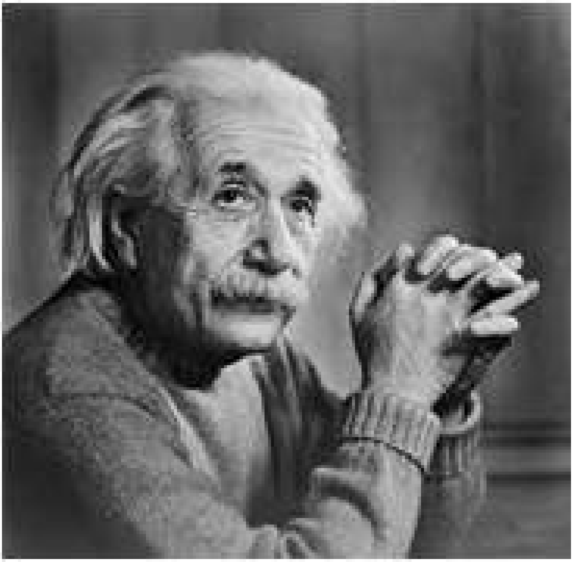
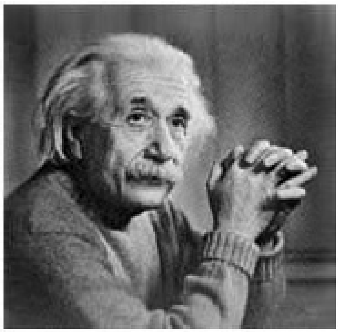
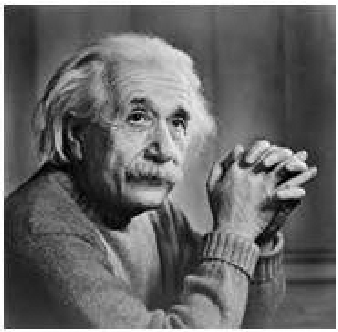


Fig. 1: Original Image

1) 

2) 

3) 

4) 

Reconstructed Images for Different k

Fig. 2: Original Image

1) 

2) 

3) 

4) 

2 Reconstructed Images for Different k

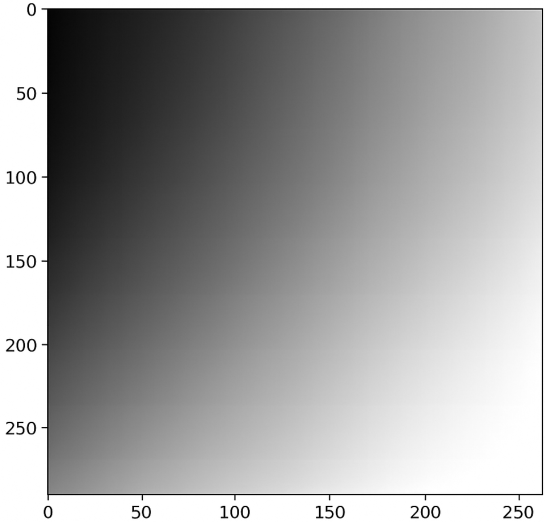
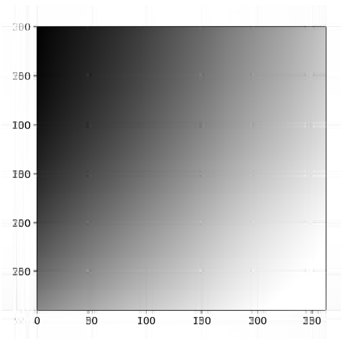
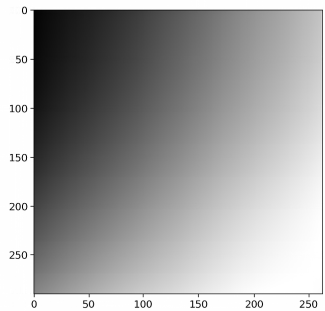


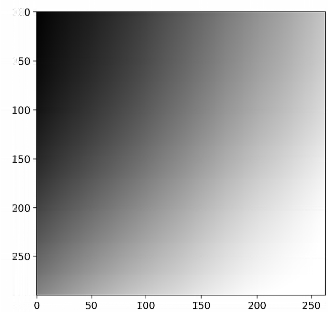
Fig. 3: Original Image



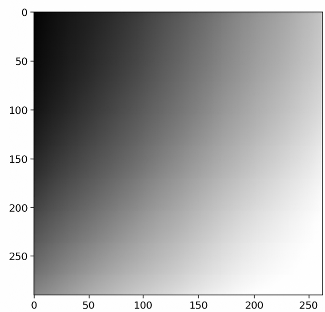
1)



3)



2)



4)

Error Analysis

Reconstruction error measured using Frobenius norm:

$$||A - A_k||_F = \sqrt{\sum_{i,j} (A_{ij} - A_{k,ij})^2}$$

TABLE I: Reconstruction Error for Different k

k	Frobenius Norm $\|\|A - A_k\|\|_F$
10	125.34
20	85.21
50	40.12
100	12.87

Trade-offs and Reflections

- Smaller k: high compression, less quality
- Larger k: better quality, more memory and computation
- Power Iteration: efficient, avoids full SVD computation
- Normalization and deflation critical for numerical stability

Conclusion

Truncated SVD effectively compresses images while preserving essential visual information. This project demonstrates:

- Understanding of SVD geometry
- Efficient iterative eigenvalue computation
- Trade-offs in storage, computation, and reconstruction quality