Comparative Discussion: Algorithmic Trade-offs and Implementation Choice

Anirudh M Abhilash

November 8, 2025

Table 1: Comparison of SVD-based Image Compression Methods

Algorithm	Advantages	Disadvantages
Golub–Kahan SVD (Householder + Givens)	• Most numerically stable and accurate full SVD method. • Well-conditioned for both symmetric and non-symmetric matrices. • Produces exact orthogonal U, Σ, and V^T.	• Computationally expensive ($O(n^3)$) — impractical for large images. • Requires full reduction even when only top-k singular values are needed. • Memory-heavy due to dense transformations.
Lanczos Bidiagonalization + Implicit QR (My Implementation)	• Efficient truncated SVD: $O(kn^2 + k^3)$ with minimal accuracy loss. • Uses only matrix–vector products — memory-light and scalable. • Implicit QR with Wilkinson shift ensures fast, stable convergence. • Ideal for image compression where small approximation is acceptable.	• Sensitive to orthogonality loss if re-orthogonalization is skipped. • Slower convergence for tightly clustered singular values. • Slight numerical drift compared to full SVD, but visually negligible. • Yields only leading singular vectors, not a full decomposition.
Jacobi Algorithm	• Extremely accurate and numerically stable. • Produces well-orthogonalized singular vectors naturally. • Useful for small matrices or benchmark-level precision.	• Very slow convergence ($O(n^3)$). • Inefficient for high-resolution image matrices.

Continued on next page

Table 1 – continued from previous page

Algorithm	Advantages	Disadvantages
Power / Subspace Iteration	• Simple to implement conceptually. • Works well for computing dominant singular values. • Requires minimal memory.	• Converges slowly for clustered or similar singular values. • Needs multiple passes over data. • Yields only leading singular vectors, not a full decomposition.
Randomized SVD (rSVD)	• Extremely fast for very large or sparse datasets. • Reduces computation via random projection and oversampling.	• Complicated to implement from scratch; requires knowledge beyond classical numerical linear algebra. • Accuracy depends on data distribution and random seed — non-deterministic output.