A4: Image Compression via Block-wise SVD

Objective

You are to explore the use of Singular Value Decomposition (SVD) for compressing grayscale images. The goal is to apply block-wise SVD, retaining only the top-k singular values in each block, and to analyze how image quality and compression ratio evolve with $k \in \{1, 2, ..., 8\}$.

Tasks

1. Preprocessing

- Choose any grayscale image of your choice (recommended: 256×256 or 512×512 resolution).
- Convert it to grayscale (if needed) and crop/resize so both dimensions are divisible by 8.

2. Block-wise SVD Function

- Partition the image into **non-overlapping 8×8 blocks**.
- Implement a function `compress_block(block, k)` that:
  - Applies SVD to the block.
  - Reconstructs it using only the top-k singular values.
  - Returns the reconstructed block.
- Carefully handle recombining blocks into a final image.

3. Compression Analysis

For each $k \in \{1, ..., 8\}$:

- Apply the block-wise SVD compression.
- Compute the **compression ratio**:
  Compression Ratio = Original Data per block / Data retained per block after top-k
- For an 8×8 block:
  - Original = 64 values
  - With top-k SVD: `k*(8+8+1)` values (U: 8×k, Σ: k, V^T: k×8)
- Save the compressed and reconstructed image for each k.

4. Visualization

- Plot **Compression Ratio vs. k**
- Plot **Reconstruction Error (Frobenius norm) vs. k**
  - **Optional:** For visual quality comparison, try computing PSNR (Peak Signal-to-Noise Ratio).

Submission Requirements

- Create a Github project that includes the following:

    - Name of your repository: MATH/CSCI485_Spring25_<Firstname>_<Lastname>
    - Within your repository, create project or folder: Assignment_5
    - Within your project, include the following:
        - Jupyter Notebook
            - Source code
            - Execution result of your code that generates required outputs, graphs and tables etc.
            - Brief description of your code and analysis of execution results as Markdown in the notebook.
        - A PDF file of the report
            - Analysis of results
            - Implementation summary
            - Important code snippets
- Submission:
    - In Canvas Assignments, submit:
        - PDF report **as file upload**
        - Link to your Github repo/projects **as URL**
    - Make sure you have invited me to your github repo/projects
- Due: as shown in Canvas