**Project Title : Simple JPEG Codec Implementation**

**Description**

This project demonstrates the development of a simplified JPEG codec to explain the fundamentals of image compression. It covers the encoding process, including color space transformation, Discrete Cosine Transform (DCT), quantization, and Huffman coding, and their reverse decoding process. MATLAB was used for the prototype implementation, and Python was employed for functional programming.

**Features**

1. **Image Compression:**
   * Converts images from RGB to YCbCr for luminance-chrominance separation.
   * Block-wise application of DCT for frequency domain transformation.
   * Quantization to reduce precision for high-frequency components.
   * Zigzag scanning and Huffman coding for entropy compression.
2. **Image Reconstruction:**
   * Decodes Huffman-coded data.
   * Applies inverse operations (IDCT, dequantization) to reconstruct the image.
   * Converts YCbCr back to RGB.
3. **Performance Metrics:**
   * Measures Peak Signal-to-Noise Ratio (PSNR) to evaluate quality.
   * Compression ratio for storage efficiency.

**Implementation Details**

1. **Languages:**
   * MATLAB: Used for algorithm exploration and prototyping.
   * Python: Final implementation utilizing libraries such as NumPy and Pillow.
2. **Workflow:**
   * **Encoding:** Includes color space conversion, DCT, quantization, zigzag scanning, and Huffman coding.
   * **Decoding:** Reverse operations to reconstruct the compressed image.
3. **Challenges:**
   * Maintaining stability in DCT operations.
   * Optimizing performance for large images.

**How to Use**

1. **Dependencies:**
   * Python libraries: NumPy, Pillow.
   * MATLAB for initial prototyping (optional).
2. **Steps:**
   * **Compression:**
     + Provide an RGB or grayscale image.
     + Run the encoding script to generate compressed data.
   * **Decompression:**
     + Use the decoding script to reconstruct the original image.
3. **Testing:**
   * Compare original and reconstructed images visually and using PSNR.

**Results**

* **PSNR Performance:**
  + MATLAB: ~36.314 dB.
  + Python: Similar performance confirmed during testing.
* **Visual Quality:**
  + Minimal artifacts with preserved overall structure and colors.
* **Compression Efficiency:**
  + Significant reduction in file size with minor loss in image quality.

**Limitations**

* Lossy compression leads to minor artifacts, particularly in high-frequency areas.
* Performance bottlenecks for large images or high-resolution scenarios.

**Future Work**

1. Implement progressive JPEG for better user experience.
2. Explore advanced entropy coding methods like arithmetic coding.
3. Introduce adaptive quantization for dynamic compression.
4. Optimize using multi-threading or GPU acceleration.