Assignment 2

Signal Processing in Practice

Tejash More
ID: 27077

January 21, 2026

1 Introduction

The Discrete Cosine Transform (DCT) is a fundamental technique in digital signal processing, particularly used for its energy compaction properties, such as in JPEG image compression. Unlike the DFT, the DCT operates on real-valued signals; therefore, it avoids generating complex coefficients, making it computationally efficient for image processing.

This project aims to:

1. Construct and visualize 1D DCT basis functions.
2. Verify the orthonormality of the DCT transform matrix.
3. Implement a block-based 2D DCT compression pipeline.
4. Analyze the trade-off between image quality (PSNR) and compression (Sparsity) using a hyper-parameter quantizer.
5. Visualize the energy compaction of the DCT.

2 Methodology and Implementation

2.1 1D Discrete Cosine Transform

The 1D DCT for a length-N signal $x[n]$ is defined by the basis functions $\phi_k[n]$:

$$\phi_k[n] = \alpha_k \cos\left(\frac{\pi(2n+1)k}{2N}\right), \quad n, k = 0, \ldots, N-1 \tag{1}$$

where the normalization factor α_k is given by:

$$\alpha_k = \begin{cases} \sqrt{\frac{1}{N}} & \text{if } k = 0 \\ \sqrt{\frac{2}{N}} & \text{if } k \neq 0 \end{cases} \tag{2}$$

2.1.1 Implementation

In the provided 'DCT.py', the function `generateDCTbasis(k, n, N)` implements this formulation. A transformation matrix $D \in \mathbb{R}^{N \times N}$ is constructed where the entry $D_{k,n} = \phi_k[n]$.

Orthonormality Verification: To verify the basis is orthonormal, we compute the Frobenius norm of the deviation from the identity matrix:

$$\text{Error} = \|DD^T - I\|_F \tag{3}$$

Theoretically, DD^T should equal the identity matrix I for an orthonormal basis.

2.2 2D DCT and Block Processing

For image compression, we extend the transform to 2D. The 2D DCT basis functions are separable products of 1D basis functions:

$$\Phi_{u,v}[m,n] = \phi_u[m] \cdot \phi_v[n] \tag{4}$$

The image is divided into non-overlapping 8×8 blocks. For each block B, the DCT coefficients C are computed via:

$$C[u,v] = \sum_{m=0}^{7} \sum_{n=0}^{7} B[m,n]\Phi_{u,v}[m,n] \tag{5}$$

2.3 Quantization Strategy

Compression is achieved by quantizing the DCT coefficients. We implement a parametric scalar quantization matrix Q, defined as:

$$Q[u,v] = 1 + s(u+v) \tag{6}$$

where $s > 0$ is a scalar parameter controlling the compression strength. The quantization operation is:

$$\hat{C}[u,v] = \text{round}\left(\frac{C[u,v]}{Q[u,v]}\right) \tag{7}$$

Reconstruction is performed by scaling back: $\tilde{C}[u,v] = \hat{C}[u,v] \cdot Q[u,v]$.

3 Analysis and Observations

3.1 Part 1: 1D DCT Basis Functions

We constructed the DCT matrix for $N = 32$, shown in fig: 1. The rows represent increasing frequency modes from top (DC component) to bottom (High Frequency).

Numerical Verification: The implementation computed the Frobenius norm $\|DD^T - I\|_F$.

- **Observed Value:** 1.8876×10^{-14}
- **Analysis:** This value is extremely close to zero, effectively representing machine epsilon (floating-point precision limits). This numerically confirms that the constructed DCT basis functions form an orthonormal set.

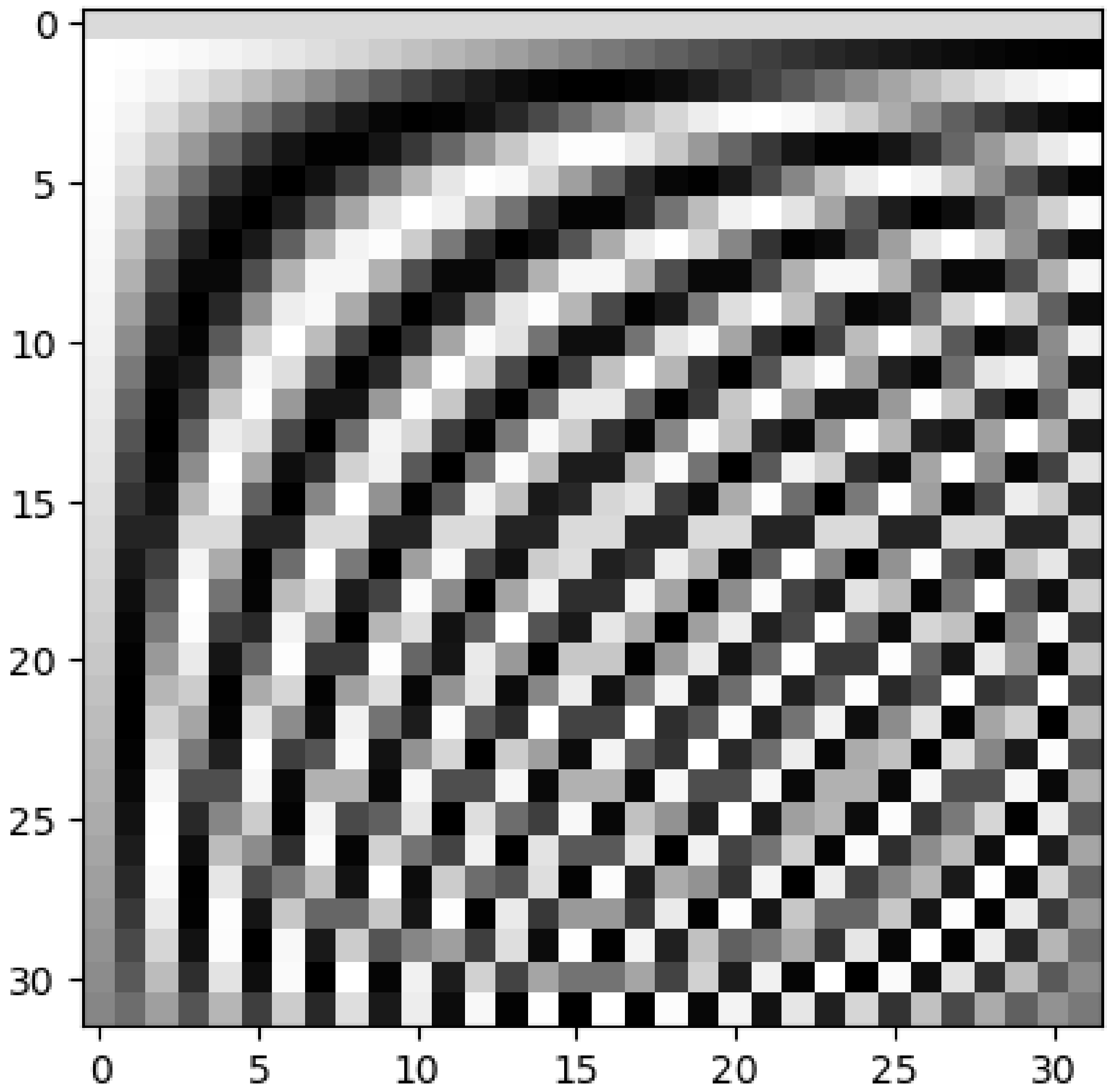

Figure 1: Visualization of the 32×32 DCT Transformation Matrix.

3.2 Visualization of 1D DCT Basis Functions

Visualization of the first $k = 8$ basis functions ($\phi_k[n]$) for $N = 32$ to analyse the cosine wave, as shown in fig: 2. $k = 0$ represents the DC component (constant value), while higher k values correspond to increasing oscillation frequencies.

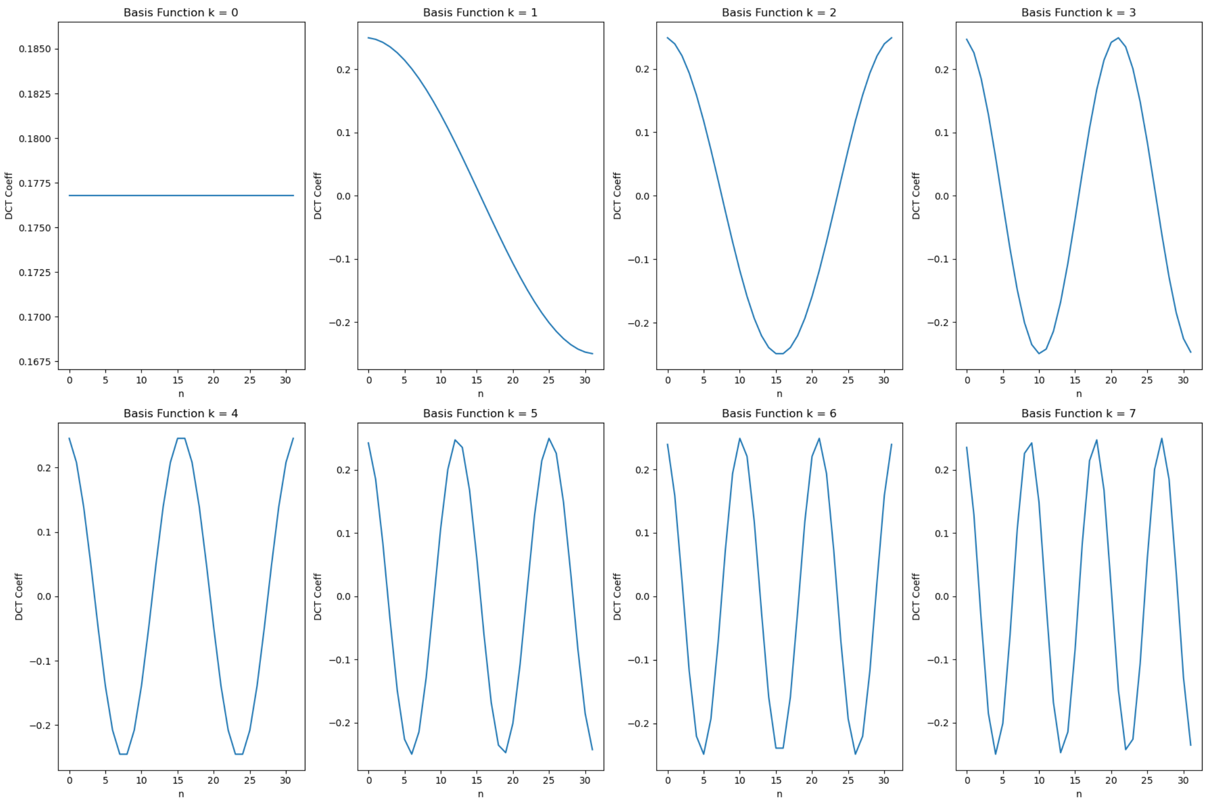

Figure 2: The first 8 DCT basis functions.

3.3 Visualization of 2D DCT Basis Functions

We extended the analysis to the 2D case by generating the basis functions for an 8×8 block. Shown in fig: 3, the 2D basis is a collection of 64 images, where each image represents a specific spatial frequency combination (u, v).

Observation: The top-left corner $(0, 0)$ represents the DC component (flat average). Moving right increases horizontal frequency; moving down increases vertical frequency. The bottom-right represents the highest frequency checkerboard pattern. This visualization confirms that the 2D DCT decomposes an image into patterns ranging from simple gradients to complex checkerboard ones. Real-world images are composed mostly of the low-frequency patterns (top-left region).

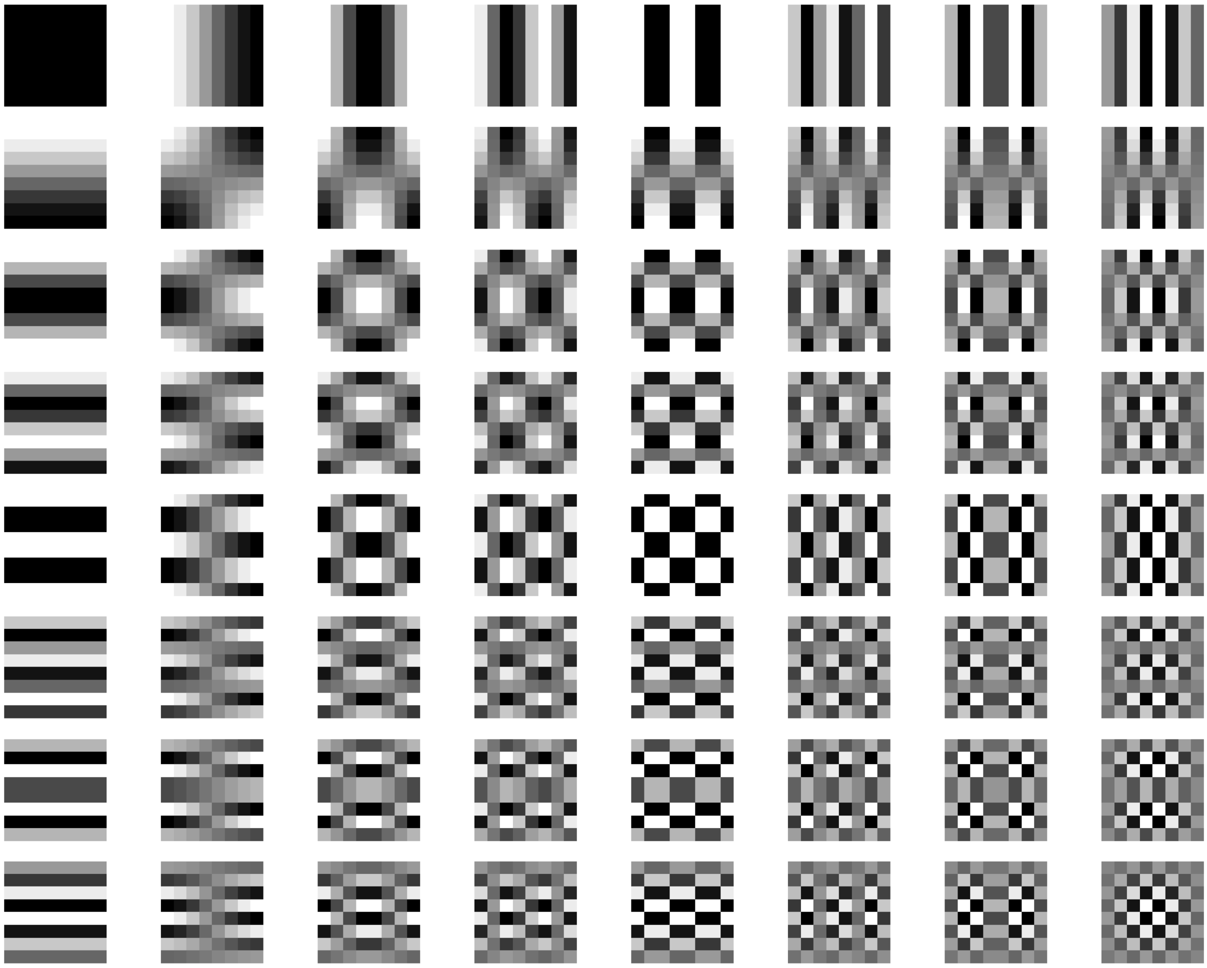

Figure 3: The 8×8 DCT Basis Images.

3.4 Verification on a Single 8×8 Block

Before processing the full image, we validated the implementation on a single 8×8 image block of an emoji.

Procedure:

1. Computed DCT coefficients C using the forward transform.
2. Reconstructed the block $\tilde{B}$ using the inverse transform (without quantization).
3. Computed the reconstruction error.

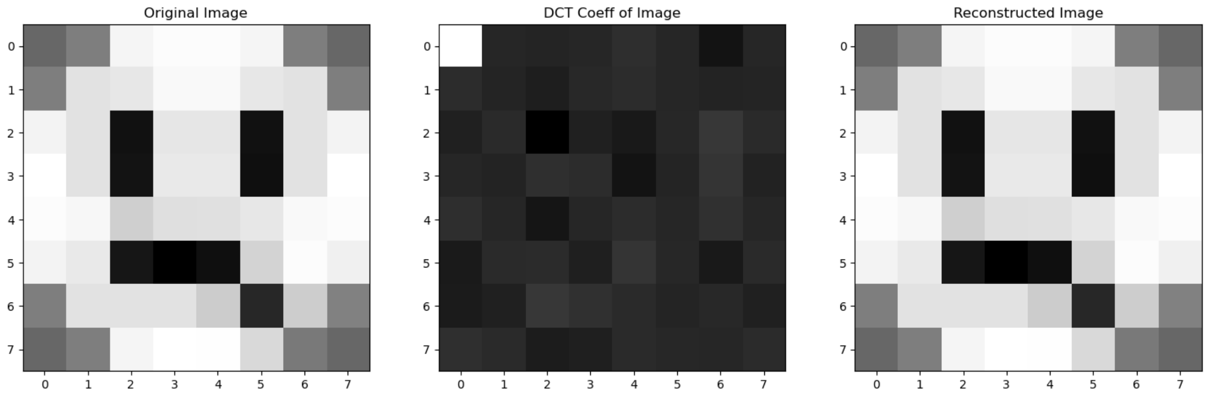

Figure 4: Visual verification of perfect reconstruction of an 8x8 Image.

Result: As shown in fig: 4, the maximum absolute error between the original and reconstructed block was:

$$\text{Error}_{max} = \max |Original - Reconstructed| \approx 8.242 \times 10^{-13} \tag{8}$$

This result (effectively zero) confirms the correctness of the forward and inverse 2D DCT implementations before moving to the full compression pipeline.

3.5 Part 2: Image Compression and Reconstruction

An image *cameraman.tif* with size 256×256 is processed in 8×8 blocks.

3.5.1 Reconstruction without Quantization

Before applying quantization, the image was transformed and immediately inverse-transformed, as shown in fig: 5.

- **Max Absolute Error:** $= 1.278 \times 10^{-12}$ (This result is considered as 0 because of computer's floating point precision limits)
- **Reasoning:** Since there is 0 loss, in the absence of quantization steps, the operation is perfectly reversible up to floating-point rounding errors.

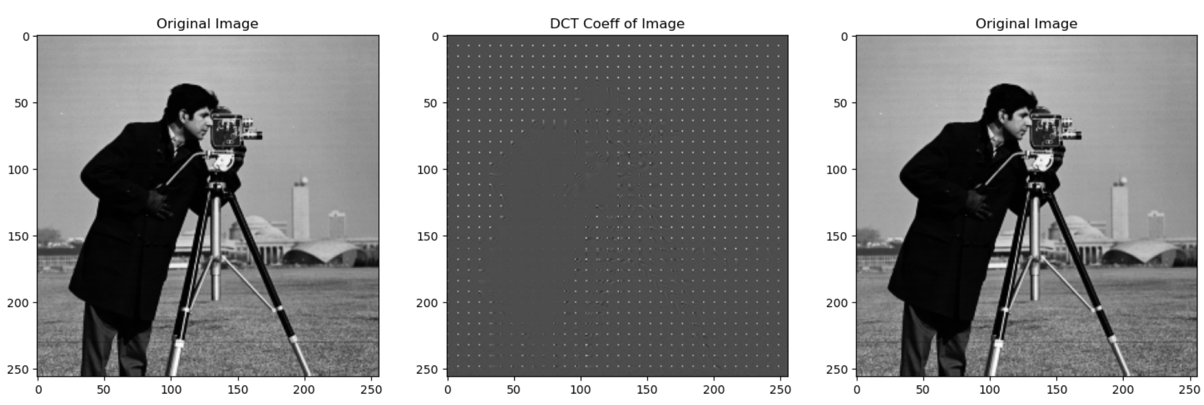

Figure 5: Visual verification of perfect reconstruction of an 256x256 Image.

3.5.2 Quality vs. Compression Study

We tested the compression strength parameter s at 5 different values, and the values are recorded in Table 1 and shown in Fig 6. As s increases, the quantization step sizes in $Q[u, v]$ increase, particularly for high-frequency components (where $u + v$ is large).

Strength (s)	Sparsity (% Zeros)	PSNR (dB)
1	0.611	42.401
3	0.769	36.026
6	0.849	32.420
12	0,907	29.457
24	0.942	27.061

Table 1: Quantitative analysis of reconstruction quality (Sparsity & PSNR) vs compression strength.

Theoretical Reasoning for Observations:

1. **Sparsity vs s:** Shown in Fig 7, As s increases, the denominator $Q[u, v]$ becomes larger. This causes small high-frequency coefficients to be rounded to zero. Consequently, sparsity increases monotonically with s.
2. **PSNR vs s:** Shown in Fig: 7, PSNR is inversely proportional to the Mean Squared Error. As sparsity increases, we discard more information (high-frequency details), leading to higher error and lower PSNR.
3. **Visual Artifacts:** Shown in Fig: 6, At high values of s, the reconstructed images exhibit "blocking artifacts." This occurs because each 8×8 block is quantized independently. The loss of coefficients causes discontinuities at the block boundaries, which the eye perceives as a grid pattern.

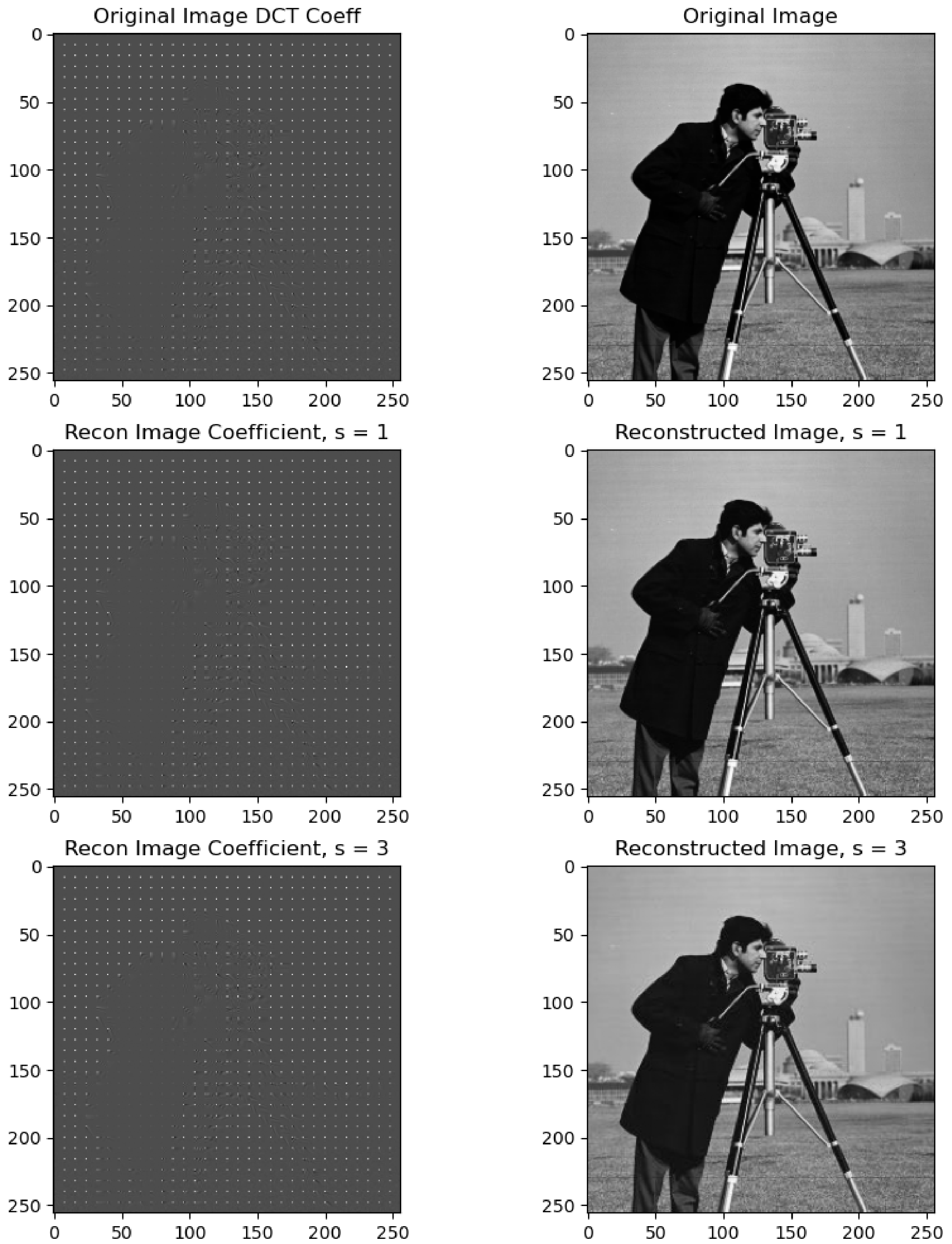

Figure 6: Comparison of Original vs. Compressed Image with different Compression Strengths.

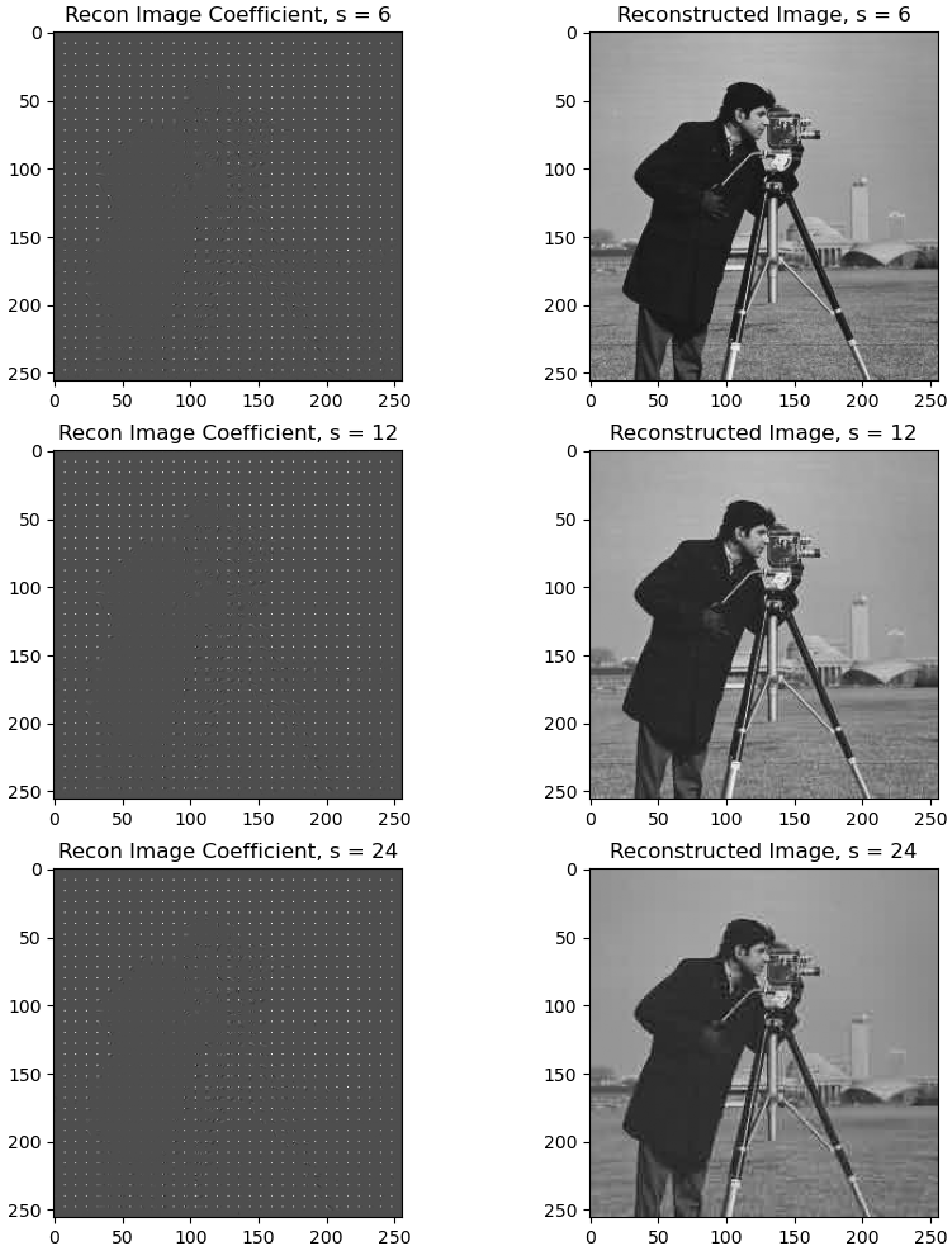

Figure 6: (Continued) Comparison of Original vs. Compressed Image with different Compression Strengths.

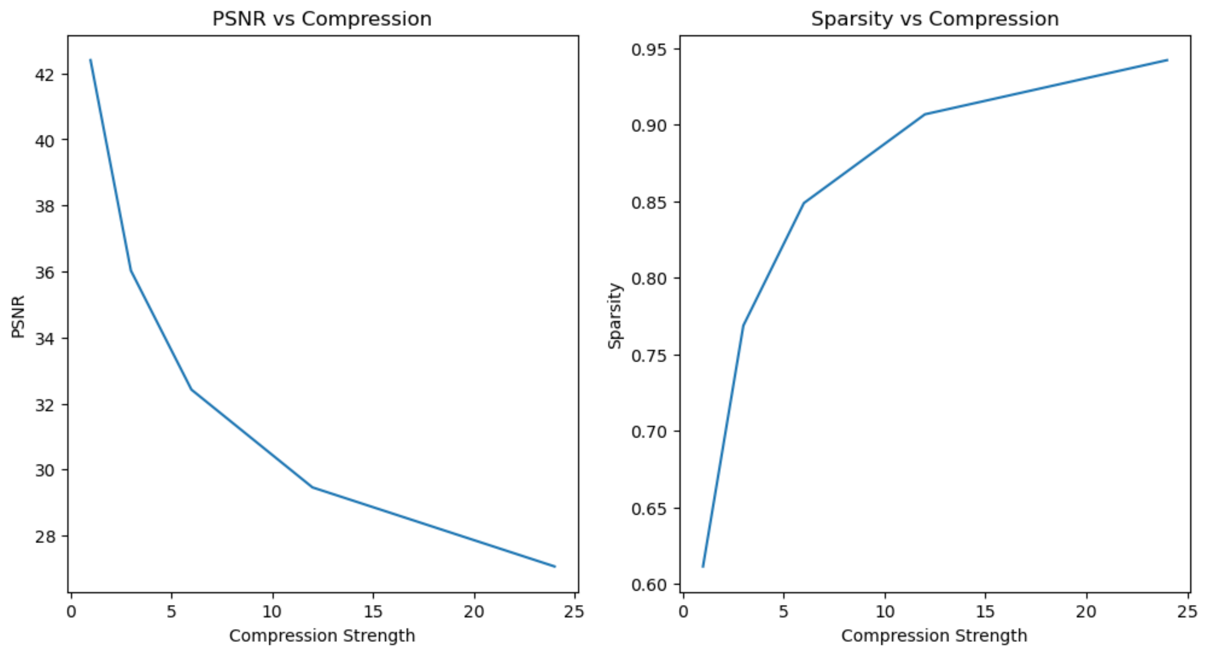

Figure 7: Plot: (Left) PSNR vs Compression Strength, (Right) Sparsity vs Compression Strength

3.6 Energy Compaction Analysis

We analyzed how much energy is contained within the top-left $K \times K$ coefficients of the DCT blocks. The energy fraction was computed as:

$$E_{ratio}(K) = \frac{\sum_{(u,v)\in S_K} |C[u,v]|^2}{\sum_{\text{all } u,v} |C[u,v]|^2} \tag{9}$$

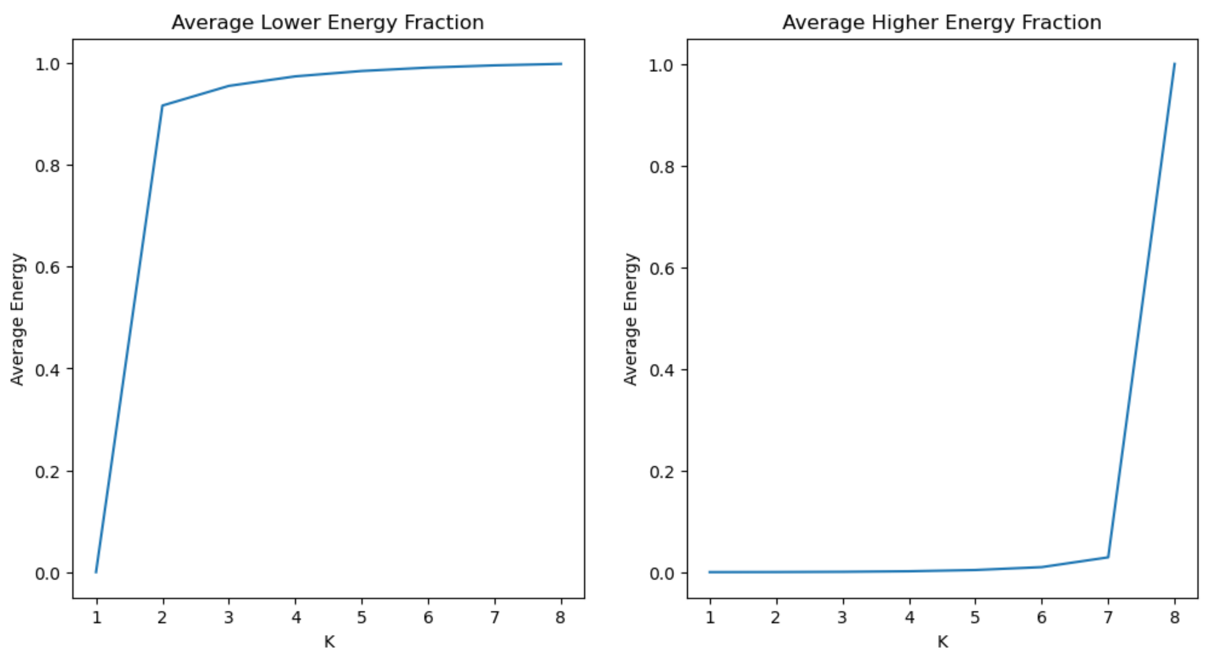

Figure 8: Energy Compaction Curve. (Left) Top-left K blocks, (Right) Bottom-right K blocks

Reasoning: Natural images are dominated by low-spatial frequencies (smooth regions). The DCT bases corresponding to low frequencies (small u, v) capture this information. The plot in Fig 8 shows a steep rise, often reaching $> 90\%$ energy with only a small K (in the plot, its $K = 2$). This "compaction" is what allows JPEG to discard high-frequency coefficients (high u, v) with little perceptual loss.

Similarly, it's observed in Fig 8 that until the bottom-right $K = 7$ blocks, energy doesn't increase much, implying that there is little to no information in those blocks. This property is used in image compression, where high-frequency DCT coefficients are removed, and the image is reconstructed using only the low-frequency DCT coefficients.

4 Conclusion

This study successfully demonstrated the utility of the Discrete Cosine Transform in image compression.

- We verified the mathematical properties of the DCT basis (orthonormality).
- We implemented a full compression pipeline involving block transformation, parametric quantization, and reconstruction.
- Our analysis confirmed that increasing quantization scaling s leads to higher sparsity (better compression) at the cost of lower PSNR (quality), eventually resulting in visible blocking artifacts.
- The energy compaction analysis highlighted why DCT is superior for image compression: significant image information is concentrated in a very small subset of coefficients.