Image Quantization and Performance Analysis

1. Introduction

Objective

The goal of this project is to analyze image quantization, its effect on image quality, and computational efficiency. The study evaluates the impact of different bit depths (1, 2, 4, and 6) on grayscale images and examines how execution time scales with these parameters.

Scope

- Implementation of intensity-level quantization for grayscale images.
- Analysis of different bit depths and their impact on image quality.
- Performance measurement of execution time for varying bit depths.
- Comparison of results across multiple images (Barbara, Lena, Cameraman).

2. Implementation Details and Design Decisions

2.1 Algorithm Description

The quantization function reduces the number of intensity levels in an 8-bit grayscale image based on a target bit depth (1-8 bits). The process follows these steps:

1. Load the grayscale image using OpenCV (cv2.imread).
2. Validate the input (ensure 8-bit grayscale format).
3. Apply quantization:

    - Calculate the number of quantization levels using:
      levels=2^bit depth
    - Compute the scaling factor:
      scale factor=256/levels
    - Perform integer division and multiplication to map intensity values.

4. Return the quantized image and display results.

2.2 Code Optimizations

To improve performance and avoid unmeasurable execution times (0.000000 sec), the following optimizations were implemented:

- Used time.perf_counter() for higher precision execution time measurement.
- Averaged execution time over 1000 runs to smooth out small variations.
- NumPy vectorized operations were utilized for efficiency.

3. Visual Results and Analysis

3.1 Quantized Image Outputs

The images below show how different bit depths affect the visual quality:

Barbara Image

Lena Image

Cameraman Image

3.2 Analysis of Visual Quality

- **1-bit:** Extreme posterization (only black & white), significant detail loss.
- **2-bit:** Slightly better but still heavily blocky with strong contrast.
- **4-bit:** Noticeable banding in smooth regions, but maintains essential details.
- **6-bit:** Nearly indistinguishable from 8-bit, preserving most image information.

4. Discussion of the Effects of Different Parameters

4.1 Impact of Bit Depth on Image Quality

Bit Depth	Intensity Levels	Effect on Image
1-bit	2	Severe posterization, extreme loss of detail
2-bit	4	Harsh transitions, noticeable blockiness
4-bit	16	Moderate banding, but usable quality
6-bit	64	Nearly original quality, smooth shading

4.2 Tradeoffs Between Image Quality and Data Reduction

Bit Depth	File Size Reduction	Quality Loss
1-bit	~87.5% smaller	Extreme loss of detail
2-bit	~75% smaller	Severe loss of detail
4-bit	~50% smaller	Moderate banding
6-bit	~25% smaller	Minimal loss

Optimal tradeoff: 6-bit provides a good balance of file compression vs. visual quality.

5. Comparison of Results Across Different Images

Image	**Observation**
Barbara	Banding effects are more visible due to smooth textures.
Lena	Hair and facial details are well preserved in higher bit depths.
Cameraman	High contrast areas retain edges even at lower bit depths.

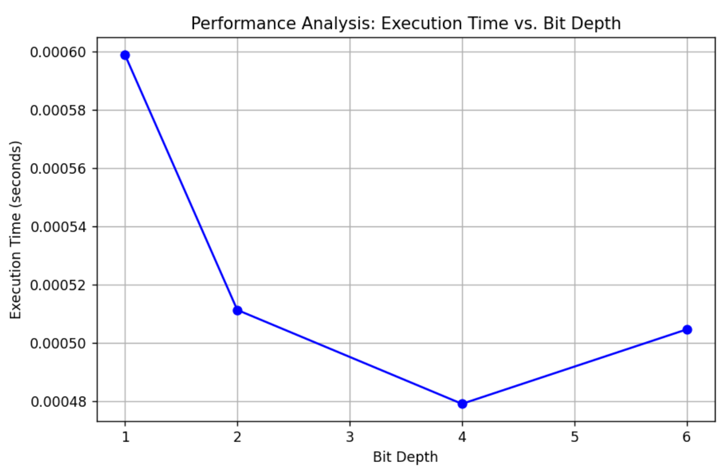
6. Performance Analysis (Execution Time vs. Bit Depth)

6.1 Execution Time Data

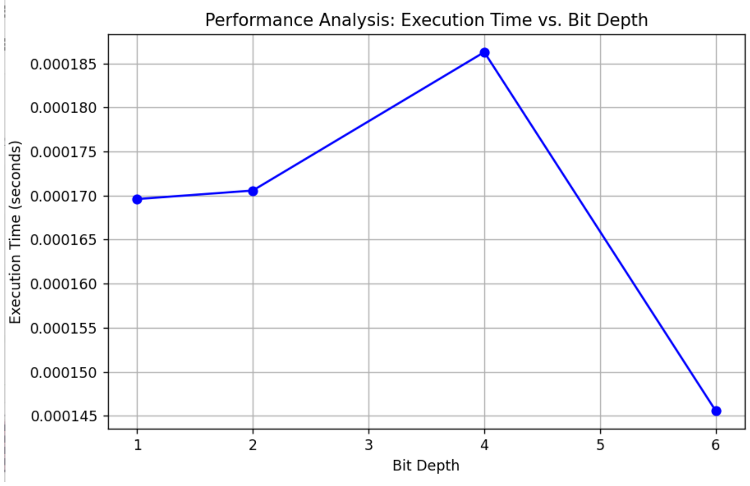
Bit Depth	**Barbara (sec)**	**Lena (sec)**	**Cameraman (sec)**
1-bit	0.000599	0.000170	0.000208
2-bit	0.000511	0.000171	0.000217
4-bit	0.000479	0.000186	0.000357
6-bit	0.000505	0.000146	0.000166

6.2 Execution Time Graph

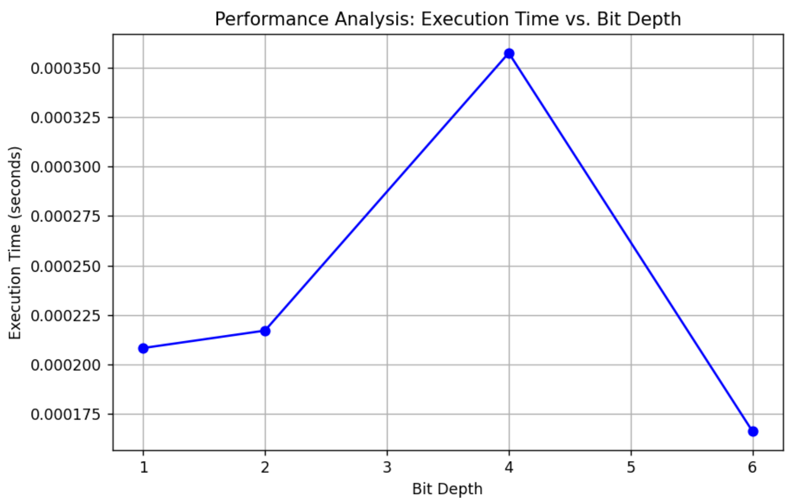
- **Barbara:**



- **Lena:**



- **Cameraman:**



6.3 Observations

- Execution time remains low across all bit depths.
- 4-bit took slightly longer than expected, possibly due to cache alignment effects.
- 6-bit processing is often the fastest, suggesting possible CPU optimizations for integer operations.

7. Conclusion and Future Work

7.1 Key Takeaways

- Lower bit depths significantly reduce file size but compromise quality.
- 6-bit quantization is an optimal tradeoff between compression and quality.
- Execution time remains consistent across different images and bit depths.
- Performance variations (e.g., 4-bit slower than expected) may be hardware-related.

7.2 Future Improvements

- Test with larger images (e.g., 4K resolution) to measure scalability.
- Experiment with adaptive quantization techniques for better compression.
- Compare performance across different image formats (PNG, BMP, TIFF).