Unit - 4: Image Compression
Assignment-2

Submission Deadline: 2 November, 11:59 pm

Part A: Theory (11 questions)

Part B: Coding

Submission Requirements: Your GitHub repository should contain the following three files:

1. A PDF or Word document with answers to the theory questions.
2. A Jupyter Notebook (.ipynb) with your coding solutions.
3. A PDF version of your Jupyter Notebook.

Note: GitHub repository should be public.

Submission Form - https://forms.gle/FekfYWe238iAoKEr5

Part A: Theory

1. Explain the need for image compression in multimedia applications. How does compression impact storage and transmission efficiency?

2. What is redundancy ? Explain three types of Redundancy.

3. Define coding redundancy. Provide examples of how coding redundancy is used to reduce image file sizes.

4. Discuss inter-pixel redundancy and how it is exploited in image compression algorithms. Provide examples of common methods to reduce inter-pixel redundancy.

5. Compare and contrast lossy and lossless image compression techniques. Provide examples of when each type of compression is more appropriate.

6. Explain Compression Ratio with an Example. What other metrics helps in understanding the quality of the compression.

7. Identify Pros and Cons of the following algorithms-
I. Huffman coding,
II. Arithmetic coding,
III. LZW coding,

IV. Transform coding,
V. Run length coding

8. Perform Huffman coding on a given set of pixel values. Show the step-by-step process and calculate the compression ratio achieved.

9. Explain the concept of arithmetic coding and how it differs from Huffman coding. Why is arithmetic coding considered more efficient in some cases?

10. Provide an example of LZW coding on a simple sequence of image pixel values.

11. What is transform coding? Explain how it helps in compressing image data by reducing redundancies in the frequency domain.

9. Discuss the significance of sub-image size selection and blocking in image compression. How do these factors impact compression efficiency and image quality?

10. Explain the process of implementing Discrete Cosine Transform (DCT) using Fast Fourier Transform (FFT). Why is DCT preferred in image compression?

11. Describe how run-length coding is used in image compression, particularly for images with large areas of uniform color. Provide an example to illustrate your explanation.

Part B : Coding

1. Implement functions for encoding and decoding an image using the following methods:

A. Transform Coding (using DCT for forward transform)
B. Huffman Encoding
C. LZW Encoding
D. Run-Length Encoding
E. Arithmetic Coding

For each method, display the **Compression Ratio** and calculate the **Root Mean Square Error (RMSE)** between the original and reconstructed image to quantify any loss of information.

Resources -

Getting Started - The Hitchhiker's Guide to Compression

Categories | Mark Nelson

LZW - Scaler Blog

Huffman Coding Algorithm

How does Transform Coding Works?

Understanding DCT and Quantization in JPEG compression - DEV Community

Huffman Encoding [explained with example and code]