Project

It is required to use Matlab, or any other programing language, to perform the following task:

Simple Image Compression

Background:

The 1D Fourier analysis represents the signal as a weighted sum of sinusoidals or complex exponentials of different frequencies. Similarly, the 2D discrete cosine transform (2D DCT) represents the 2D signal (e.g. an image or a block of an image) as a weighted sum of basis images of different spatial frequencies, as shown in Fig. 1. Spatial frequencies refer to the rate of variation of pixel intensity values with respect to space coordinates.

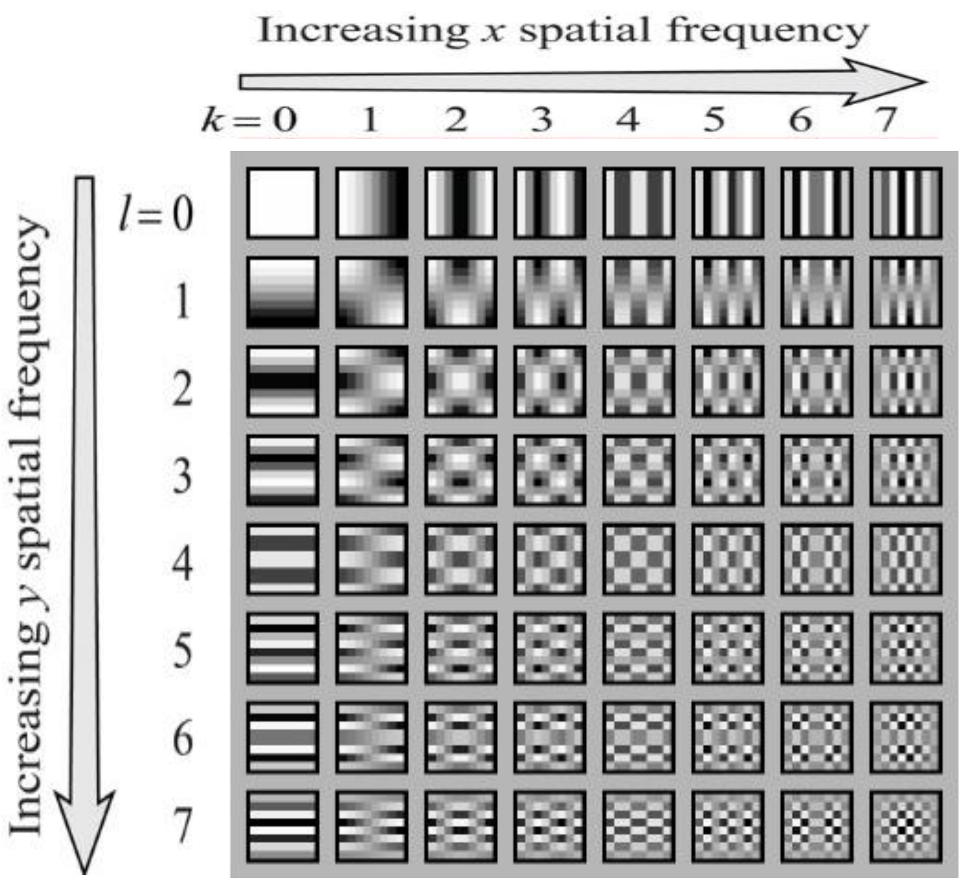

Fig. 1: 2D DCT basis images

The top left corner is DC. Images towards the top left have low horizontal and vertical spatial frequencies. Images towards the right have higher horizontal spatial frequency. Images towards the bottom have higher vertical spatial frequency. The bottom right image has the maximum horizontal and vertical spatial frequencies.

Most of the information of natural images is concentrated in the lower spatial frequencies (the top left coefficients have significantly higher values). Additionally, the human eye is more sensitive to lower frequencies and is much less likely to notice the loss of very high spatial frequency components (very fine details). Therefore, ignoring higher spatial frequencies has minimal impact on the perceived image quality.

DCT is widely used in image, video, and audio compression, due to its ability to compact most of the energy of a signal into a few coefficients.

Task:
In this task, you will implement a simple image compression algorithm. You will read an input image and process each of its color components (red, green, and blue) in blocks of 8×8 pixels. Each block will be converted into frequency domain using 2D DCT and then only few coefficients are retained, while the rest are ignored.

a) Read the image file 'image1.bmp'. Extract and display each of its three color components. Repeat the following steps for $m = 1,2,3,4$:
b) To compress the image, process each color component in blocks of 8×8 pixels. Obtain the 2D DCT of each block. It has the same dimensions as the input block, corresponding to the contribution of the 64 basis images shown in Fig. 1. Retain only the top left square of the 2D DCT coefficients of size $m \times m$. That is, if the DCT coefficients are X[1:8,1:8], retain only the top left $m \times m$ coefficients, $X[1{:}m, 1{:}m]$, assuming that the top left coefficient is $X[1,1]$. The rest of coefficients are ignored.
c) Compare the size of the original and compressed images.
d) Decompress the image by applying inverse 2D DCT to the retained coefficients of each block. Assume the rest of the coefficients in each block are zeros. Display the image.
e) The quality of the decompressed image is measured using the Peak Signal-to-Noise Ratio (PSNR), which is defined by

$$PSNR = 10 \log_{10} \frac{peak^2}{MSE}$$

where $peak$ is the peak value for the pixels according to the image datatype (e.g. for uint8 image it is 255). Mean square error (MSE) between the original image and the decompressed image is obtained by subtracting the corresponding pixel values of two images and obtaining the average of the squares of all the differences. Obtain the PSNR for each value of m.
f) Plot a curve displaying the PSNR (on the vertical axis) against m (on the horizontal axis). Comment on the resulting graph and quality of images.

Deliverables:

1. One **uncompressed pdf** project report containing:
    a. Explanation of your work.
    b. All the required results and answers to questions.
    c. All the required images and figures. Label your figures properly.
    d. All the codes, included at the end.
2. One zip file containing all the codes, compressed and decompressed images for different values of m.

Instructions:

- You can work in teams of up to 2 members per team.
- Any copied results or codes will result in zero grade for both teams.
- The code in the report should be supplied as text, not as screenshots.
- You should cite any references that you use.

Due date: May 10, 2024, at 11:59 pm.