CLASS : T.E. E&TE **SUBJECT: DIP**

EXPT. NO. : 8 **DATE:**

TITLE : IMAGE COMPRESSION USING DCT

CO2:	Perform the image compression techniques on image.

AIM:

1. To apply Discrete Cosine Transform (DCT) on an image.
2. Reconstruct the image by taking Inverse Discrete Cosine Transform (IDCT).
3. Perform DCT based quantization on given 8X8 Matrix.

SOFTWARE REQUIREMENT: Google Colaboratory /Jupyter Notebook

8.1THEORY:

8.1.1 Discrete Cosine Transform

In transform coding, a reversible, linear transform like DFT or the discrete cosine transform (DCT) is used to map an image into a set of transform coefficients, which are then quantized and coded for most natural images, a significant number of coefficient have small magnitudes and can be coarsely quantized with a little image distortion.

DCT is a child of DFT, only cosine terms are taken.

- 1D DCT is given by,

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cdot \cos[(2x+1)u\,\Pi/2N]$$

- 1D IDCT is given by,

$$f(x) = \alpha(u) \sum_{u=0}^{N-1} C(u) \cdot \cos[(2x+1)u\,\Pi/2N]$$

where, $\alpha(u) =$ sqrt(1/N) ….for $u = 0$
$=$ sqrt(2/N) ….for $u = 1,2,\ldots,N-1$

Image is a two dimensional signal f(x, y), hence in order to find DCT of an image DCT must be applied in the 2 directions x and y.

- 2D DCT is given by,

$$C(u,v) = \alpha(u)\,\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cdot \cos[(2x+1)u\,\Pi/2N] \cdot \cos[(2y+1)v\Pi/2N]$$

- 2D IDCT is given by,

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u) \cdot \alpha(v) \cdot C(u, v) \cdot \cos[(2x+1)u\,\Pi/2N] \cdot \cos[(2y+1)v\,\Pi/2N]$$

where,
$\alpha(u) =$ sqrt(1/N) ….for $u = 0$
$=$ sqrt(2/N) ….for $u = 1,2,\ldots,N-1$
$\alpha(v) =$ sqrt(1/N) ….for $v = 0$
$=$ sqrt(2/N) ….for $v = 1,2,\ldots,N-1$

Consider an image f(x, y) of size M×N , it is subdivided into sub-images of sub-size n×n and DCT of these sub-images is found out. The DCT of the sub-image is obtained by using the above formula.

8.1.2 Quantization:

Our 8*8 block of DCT coefficients is now ready for compression by quantization. A remarkable and highly useful feature of the JPEG process is that in every step, varying levels of image compression and quality are obtainable through selection of specific quantization matrices. This enables the user to decide on quality levels ranging from 1 to 100, where 1 gives the poorest image quality and highest compression. As a result, the quality/compression ratio can be tailored to suit different needs.Subjective experiments involving the human visual system have resulted in the JPEG standard quantization matrix. With a quality level of 0, this matrix renders both high compression and excellent decompressed image quality.

8.1.3 Advantages:

- Transform co-efficient are relatively uncorrelated.
- Energy is highly compacted.
- Reasonable robust relative to channel error.
- There is no loss in the data.

8.1.4 Disadvantages:

While the DCT-based image coders perform very well at moderate bit rates, at higher compression ratios, image quality degrades because of the artifacts resulting fro the block based DCT scheme.

8.1.5 Applications:

The DCT is used in JPEG image compression, MPEG, DV and Theora video compression. DCT is also widely used in signal and image processing, especially for lossy data compression, because it has a strong energy function.

8.2 Algorithm:

1. Start.
2. Take 8X8 Matrix.
3. Leveled off the matrix by subtracting 128 from each entry. (Matrix M)
4. Obtain DCT of above matrix by using the 2D DCT formula. (Matrix D)
   a. Obtain the coefficient matrix by using formula

$$T_{i,j} = \left\{ \begin{array}{ll} \frac{1}{\sqrt{N}} & \text{if } i = 0 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2j+1)i\pi}{2N} \right] & \text{if } i > 0 \end{array} \right\}$$

(Matrix T).

   b. Obtain DCT Matrix, D = TMT'.
5. Select Quantization level, and initialize standard Q matrix. (Matrix Q)
6. Obtain Quantization by dividing each element in the transformed image matrix D

by the corresponding element in the Q matrix. And rounding off to the nearest integer value. (This is quantized matrix can be further coded).

C (i,j) = round(D(i,j) / Q(i,j));

7. For Decompression, R(i, j) = Q(I, j) x C(i, j)
8. Obtain the IDCT of above matrix, and round off it to the nearest integer value.
9. Add 128 to each element.
10. Compare original and recover image.

(Note: If the M x N size image has to be compressed, divide the image into 8x8 blocks and repeat the above procedure for each block.)

8.3 Conclusion:

____

____

____

____

____

____

8.4 References:

i. Gonzalez R, Woods R, “Digital image processing”, Pearson Prentice Hall, 2008.

ii. Gonzalez R, Woods R, Steven E, “Digital Image Processing Using MATLAB®”, McGraw Hill Education, 2010.

iii. Jayaraman S, Esakkirajan S and Veerakumar T, “Digital Image Processing” Tata McGraw Hill, 2010

iv. Joshi, Madhuri A. “Digital Image Processing: an algorithm approach”, PHI Learning Pvt. Ltd., 2006.

v. Pictures taken from: http://www.imageprocessingplace.com/root_files_V3/image_databases.html

____

(Course Teacher)