LAB 7: IMAGE COMPRESSION TECHNIQUES-1: DISCRETE COSINE TRANSFORM (DCT)

The discrete cosine transform (DCT) represents an image as a sum of sinusoids of varying magnitudes and frequencies. The dct2 function computes the two-dimensional discrete cosine transform (DCT) of an image. The DCT has the property that, for a typical image, most of the visually significant information about the image is concentrated in just a few coefficients of the DCT. For this reason, the DCT is often used in image compression applications. For example, the DCT is at the heart of the international standard lossy image compression algorithm known as JPEG.

Experiment No. 37 Use discrete cosine transform to compress an image

1. Read an image into the workspace, then convert the image to grayscale.

   ```
   RGB = imread('autumn.tif');

   I = im2gray(RGB);
   ```

2. Perform a 2-D DCT of the grayscale image using the dct2 function.

   ```
   J = dct2(I);
   ```

3. Display the transformed image using a logarithmic scale. Notice that most of the energy is in the upper left corner.

   ```
   imshow(log(abs(J)),[])

   colormap parula

   colorbar
   ```

4. Remove the low frequencies from the DCT matrix by setting the last quarter to zero.

   ```
   [r,c]= size(J);

   J(r/2:r,c/2:c) = 0;
   ```

5. Reconstruct the image using the inverse DCT function idct2. Rescale the values to the range [0, 1] expected of images of data type double.

   ```
   K = idct2(J);

   K = rescale(K);
   ```

6. Display the original grayscale image alongside the processed image. The processed image has fewer high frequency details, such as in the texture of the trees.

   ```
   montage({I,K})

   title('Original Grayscale Image (Left) and Processed
   Image (Right)');
   ```

Task:

- Run the experiment and show the results.