ENGR 391: Computer Vision
Homework Assignment # 4
Spring 2024

Problem 1: DCT and its properties

The goal of this problem is to study some of the properties of the discrete cosine transform. We want to compare between the upper left corner and the bottom right corner in terms of the information contained in the DCT, and the possibility to compress the image using the DCT and its properties. We assume that the size of the original image is (n1, n2). Figure 2 illustrates the process where the top 100 by 200 elements from the top left of the DCT are kept to create the reduced size image.

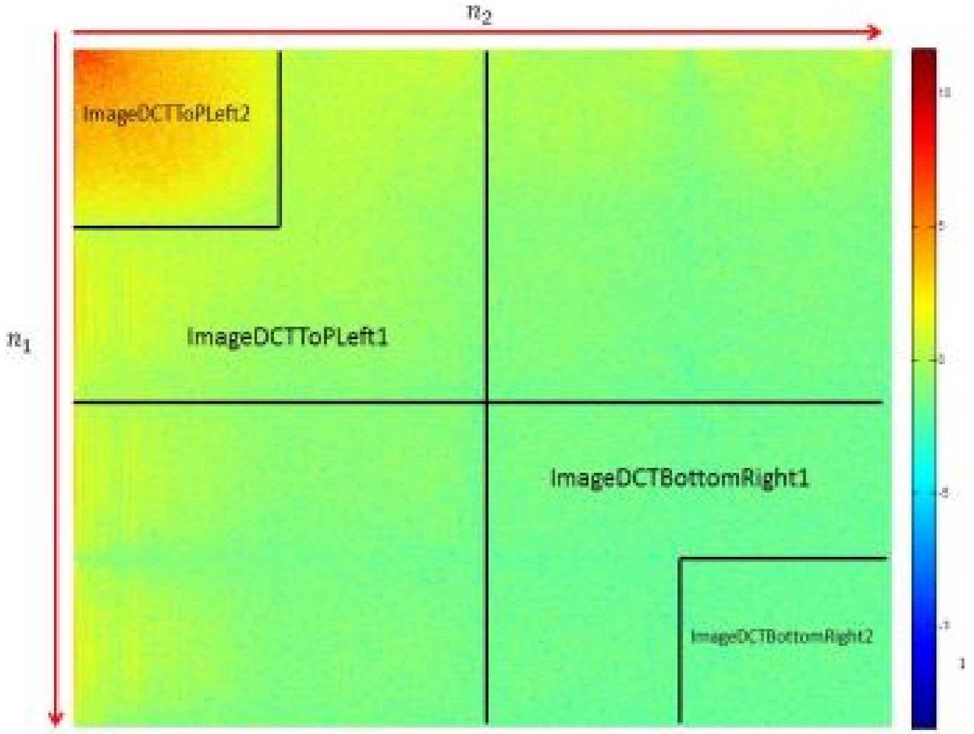

Figure 1. Ilustration of the DCT

1. Apply the DCT to an image of your choice. To display the DCT in Matlab, you can use instruction: imshow(log(abs(ImageDCT)),[]) colormap(jet), colorbar (1) where ImageDCT is the DCT of your image. You can use the log function in Python as well.

2. Now we want to reduce the size of ImageDCT by keeping only the top left corner as shown in figure 1. We do this twice:

   - ImageDCTToPLeft1: We keep (n1/2, n2/2) elements from the top left of ImageDCT.
   - ImageDCTToPLeft2: We keep (n1/4, n2/4) elements from the top left of ImageDCT.

   Write code to obtain ImageDCTToPLeft1 and ImageDCTToPLeft2 from ImageDCT.

3. Perform the inverse DCT on these matrices (ImageDCT ToPLeft1, ImageDCTToPLeft2), show and discuss your results. Hint: If you are using Matlab, you can use command imshow(K,[ ]) to display the images where K is the inverse DCT.

4. Now we keep only the bottom right corner of the matrix representing ImageDCT as shown in figure 1. Again, we do this twice:

   - ImageDCTBottomRight1: We keep (n1/2, n2/2) elements from the bottom right of ImageDCT.
   - ImageDCTBottomRight2: We keep (n1/4, n2/4) elements from the bottom right of ImageDCT.

   Write code to obtain ImageDCTBottomRight1 and ImageDCTBottomRight2 from ImageDCT.

5. Perform the inverse DCT on these matrices (ImageDCTBottomRight1, ImageDCTBottomRight2), show your results. Hint: If you are using Matlab, you can use command imshow(K,[ ]) to display the images where K is the inverse DCT.

6. Discuss and compare between the results of questions 3 and 5.

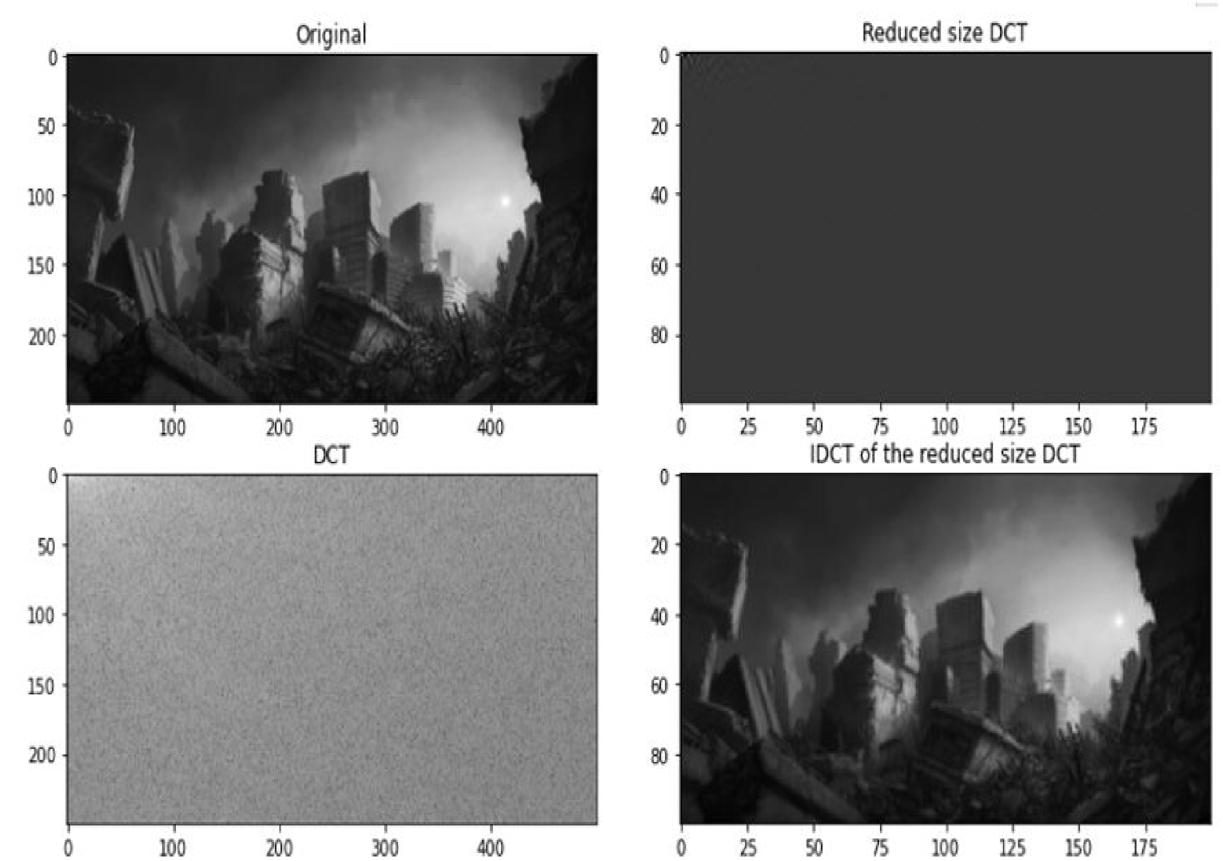

Figure 2. Ilustration of the DCT: Top left: Original image (250 x 500), Bottom left: DCT of original image (250 x 500). Top right: Reduced size DCT (100 x 200). Bottom Right: Reduced size image (100 x200)

Python hint

First convert uint8 to float type, apply the DCT and use log processing. A **suggested** approach is below.

```
img = cv2.imread('img1.jpg', 0) #Read the picture,
img1 = img.astype('float') #Convert uint8 to float type
img_dct = cv2.dct(img1) #Perform discrete cosine transform
img_dct_log = np.log(abs(img_dct)) #do log processing
img_recor = cv2.idct(img_dct) #Perform inverse discrete cosine transform
```