# DSAA Assignment 4

**Meher Shashwat Nigam, 20171062**

## Question 1

### Part 1

(a) **created mat_dct(),** which outputs the the 2D-DCT basis function F.

| Inbuilt DCT | Self Implemented DCT |
|---|---|
| 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536<br>0.4904 0.4157 0.2778 0.0975 -0.0975 -0.2778 -0.4157 -0.4904<br>0.4619 0.1913 -0.1913 -0.4619 -0.4619 -0.1913 0.1913 0.4619<br>0.4157 -0.0975 -0.4904 -0.2778 0.2778 0.4904 0.0975 -0.4157<br>0.3536 -0.3536 -0.3536 0.3536 0.3536 -0.3536 -0.3536 0.3536<br>0.2778 -0.4904 0.0975 0.4157 -0.4157 -0.0975 0.4904 -0.2778<br>0.1913 -0.4619 0.4619 -0.1913 -0.1913 0.4619 -0.4619 0.1913<br>0.0975 -0.2778 0.4157 -0.4904 0.4904 -0.4157 0.2778 -0.0975 | 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536 0.3536<br>0.4904 0.4157 0.2778 0.0975 -0.0975 -0.2778 -0.4157 -0.4904<br>0.4619 0.1913 -0.1913 -0.4619 -0.4619 -0.1913 0.1913 0.4619<br>0.4157 -0.0975 -0.4904 -0.2778 0.2778 0.4904 0.0975 -0.4157<br>0.3536 -0.3536 -0.3536 0.3536 0.3536 -0.3536 -0.3536 0.3536<br>0.2778 -0.4904 0.0975 0.4157 -0.4157 -0.0975 0.4904 -0.2778<br>0.1913 -0.4619 0.4619 -0.1913 -0.1913 0.4619 -0.4619 0.1913<br>0.0975 -0.2778 0.4157 -0.4904 0.4904 -0.4157 0.2778 -0.0975 |

(b)

**myDCT(im, F)**, which takes as input any given 8 x 8 image im and the basis matrix F. The output of this function is the DCT transformed image.

(c)

**myIDCT(im, F)**, which computes the inverse DCT transform of a  matrix im.

(d)

**myDCT_quantization(imDCT, qm, c)**, which takes as input the DCT transformed block imDCT, the quantization matrix qm and the compression factor c (divide image by c times the quantization matrix). Output is the quantized DCT image (imqDCT).

(e)

**myDCT dequantization(imqDCT, qm, c),** which de-quantizes the quantized DCT image

(f)

**RMSE(im1, im2),** which computes RMSE error between two images of arbitrary size.

(g)

**my_entropy(im),** which computes the entropy of a given image.

## Part 2

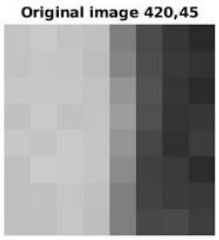
The DCT, quantized DCT and reconstructed image for the 8x8 sub-windows extracted from the LAKE image and whose top left corners are at the coordinates are shown as below :

(420, 45), (427, 298) and (30, 230). For this we have used the classical quantization matrix for luminance and c = 2.

The operations are performed on the blocks with the given starting positions.After Quantisation the number of colour components used to describe the image is reduced. This process causes us to lose information. We use less number of colours to display the same image.

As seen from the images above, the blocks after quantisation and reconstruction are a bit blurred with their details in the colour pixels reduced. However, they are pretty similar to the original image blocks.

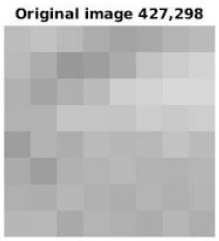
As the compression factor is increased, the number of colours used to represent the image reduces deteriorating the detail and quality of the image.


Original image 420,45


dct image 420,45


quantized dct image 420,45


Reconstructed image 420,45


Original image 427,298

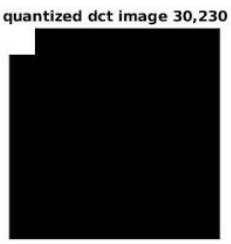

dct image 427,298

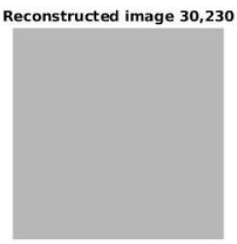

quantized dct image 427,298

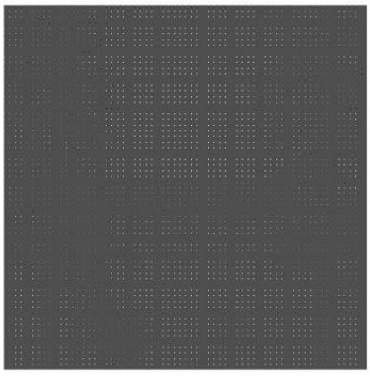

Reconstructed image 427,298


Original image 30,230

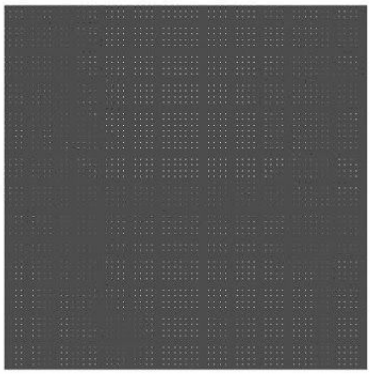

dct image 30,230


quantized dct image 30,230

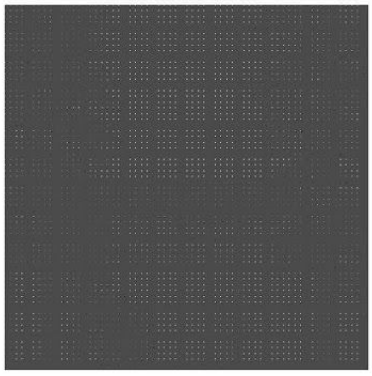

Reconstructed image 30,230



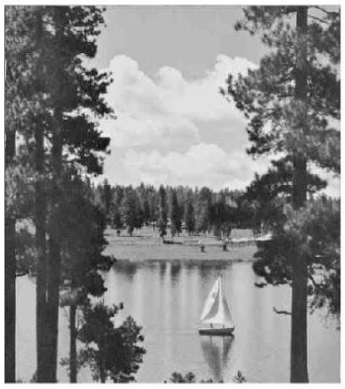
c=2



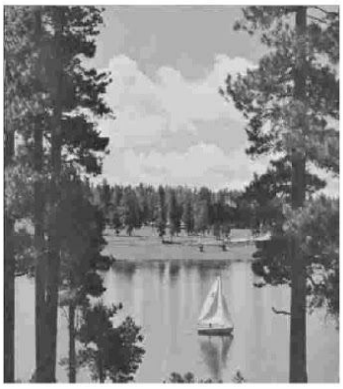
c=5



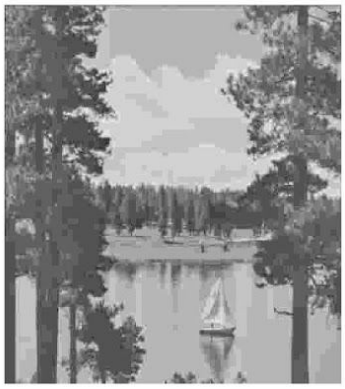
c=10



c=2



c=5



c=10

## Part 3

Applied the DCT transform (and quantization) to all 8x8 sub windows of the LAKE image and created an image with all the resulted DCT images at the same positions as their corresponding image. Ran the same for c=2,5,10.

## Part 4

Reconstructed the images from part 3. The highest value of c so that the distortions of the reconstructed image are just perceptible is around 5. This is because the image is now represented by a relatively less number of colour components than before. As c increases the change in the values upon reconstruction.

| c | entropy | rmse |
|---|---|---|
| 2 | 0.001016712266656 | 7.413158529385286 |
| 5 | 0.003487071762077 | 10.376301603198748 |
| 10 | 0.010267123938707 | 14.021737346328994 |

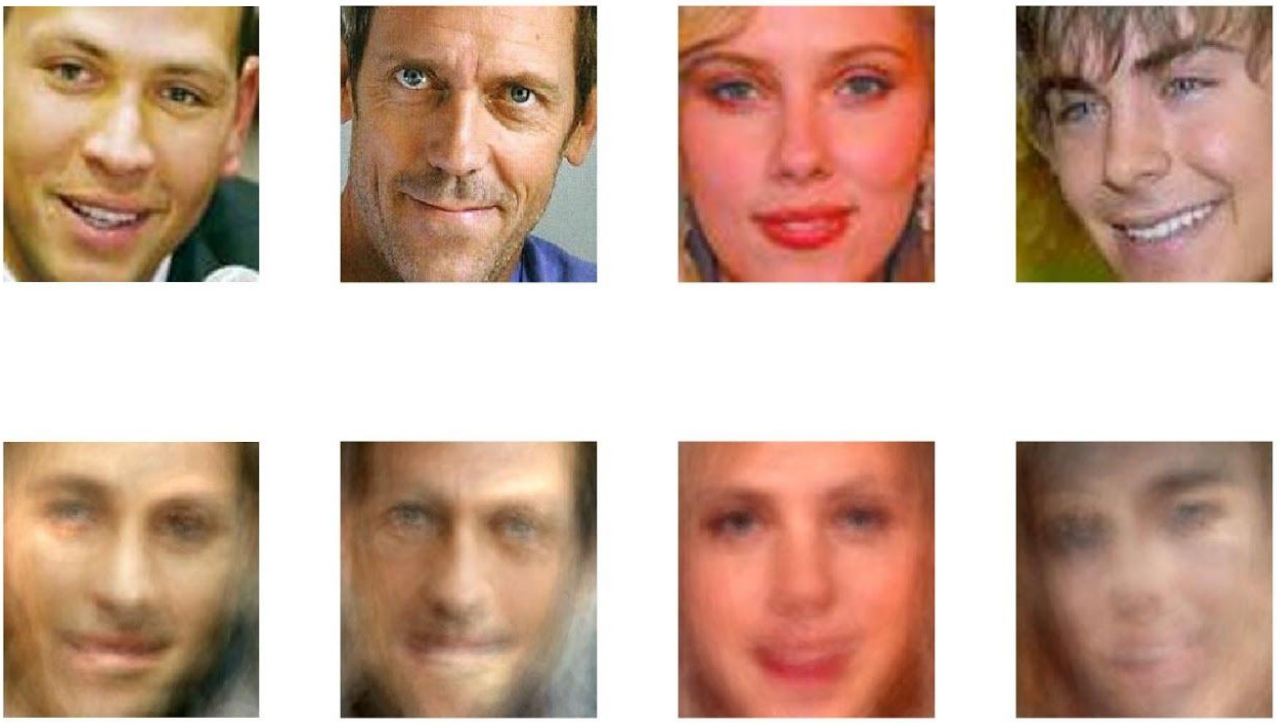
# Question 2

## Part 1

Reconstructed Images with components  35, 100 and 500 are shown below. As the number of components gets larger the image becomes more similar to the original image.

Using PCA, we can express an image as a weighted linear combination of its principal components to obtain the resultant image.

An image constructed using eigenvectors (actual image,35,100,500 eigenvectors)

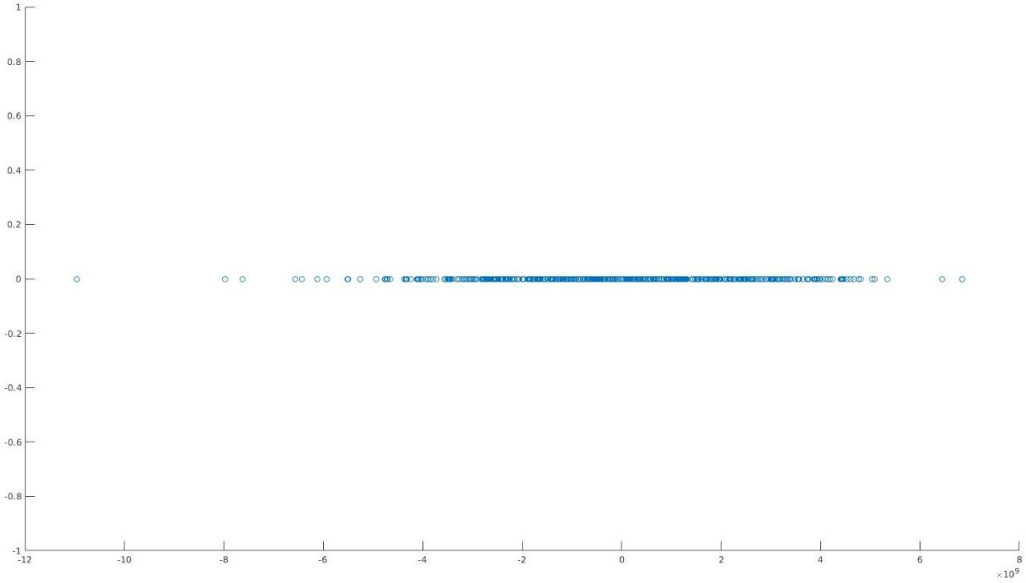


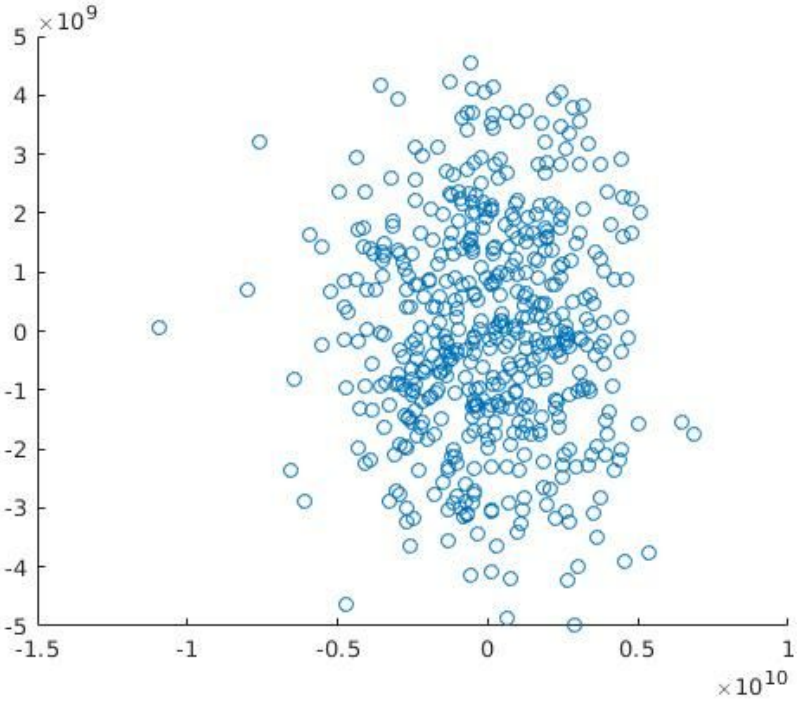
These are plots of reconstructed images using 35 eigenvectors. Part of the code has been commented ( for 100 and 500 eigenvector reconstruction).

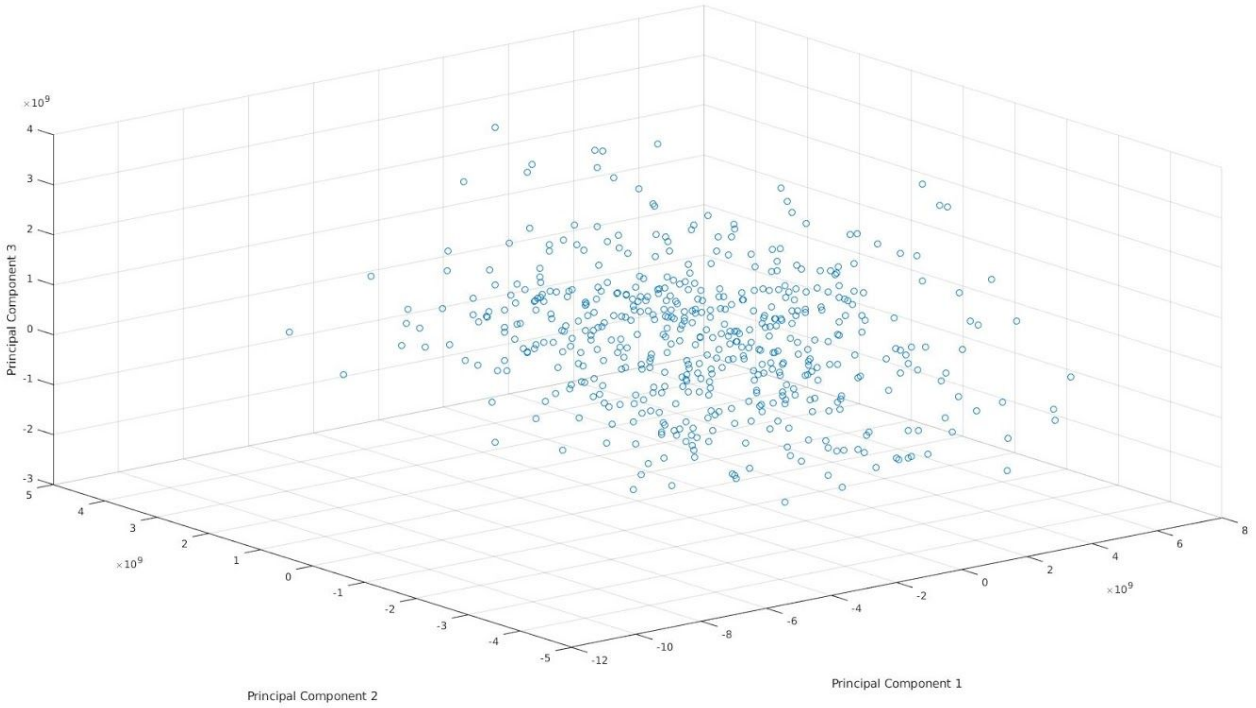
## Part 2

The scatter plots for One Dimensional, Two Dimensional and Three Dimensional for 1, 2, and 3 principal components are shown below.

All the image data was projected onto pc1,pc2 and pc3 and plotted as shown:


1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-12
-10
-8
-6
-4
-2
0
2
4
6
8
×10^9


×10^9
5
4
3
2
1
0
-1
-2
-3
-4
-5
-1.5
-1
-0.5
0
0.5
1
×10^10


×10^9
4
3
2
1
0
-1
-2
-3
Principal Component 3
5
4
3
2
1
0
-1
-2
-3
-4
-5
×10^9
Principal Component 2
-12
-10
-8
-6
-4
-2
0
2
4
6
8
×10^9
Principal Component 1