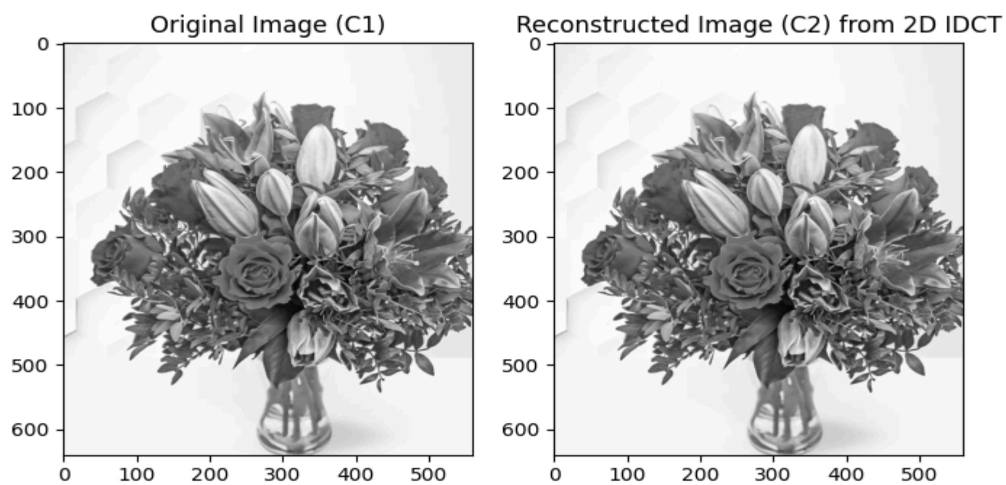
COL783: DIGITAL IMAGE ANALYSIS
Assignment 3

DHARMESWAR BASUMATARY(2020CS50423)
MUJAHID HUSSAIN(2020CH70182)

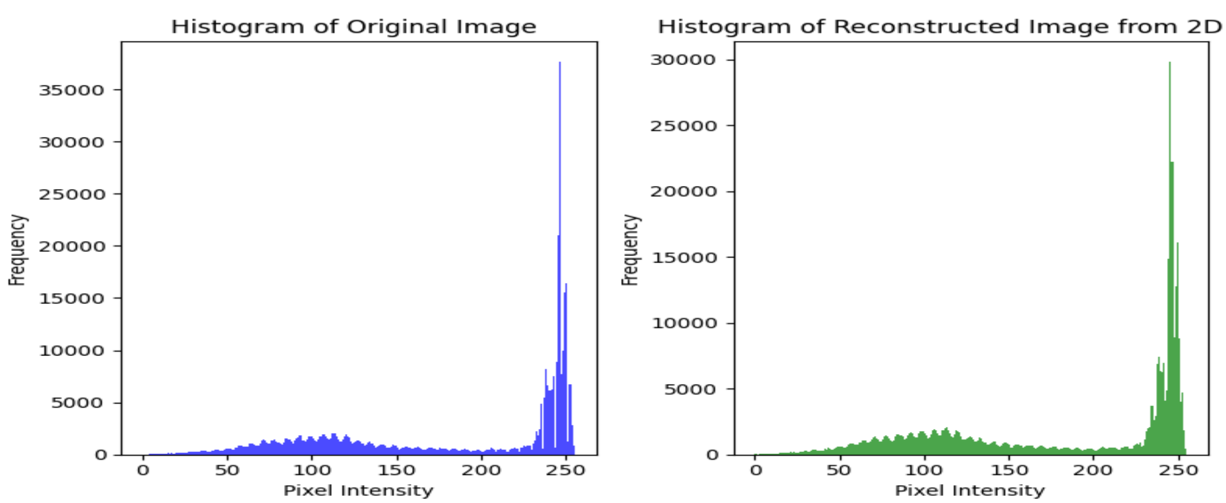
Q1.

Part a:

MSE between DCT of library and my implementation = 1.4438049e-09 (nearly
sum of absolute_error between original image t and f = 3.001083364040369e-06
Original Image:



Part B:



Part C:

MSE of reconstructed image t after retaining only the 1/16th largest (in absolute value) coefficients= 168.7435055969652

1/(MN) times the sum of squares of deleted coefficients = 168.74350559696506

Proof:

1.We have an image (or matrix) f of size M×N .

2. Let T be an orthogonal transform applied to f, resulting in coefficients

t=T(f)

3. We retain only the 1/16th coefficient largest (in absolute value) coefficients in t, setting the rest to zero to obtain an approximation t′.

Let f′ be the inverse transform applied to t′, which approximates f.

The MSE between f and f′ is:

MSE =1/MN * $\sum$ (f _ij - f'_ij) ^2 , 0<=i < M, 0<=j<N ...(1)

Since T is an orthogonal transform:

⟨T(f),T(g)⟩=⟨f,g⟩

|T(f)|^2 = |f|^2

i,j∑fij2=i,j∑tij2

$$\Sigma(f_{ij})^2 = \Sigma(t_{ij})^2$$

deleted t_ij, ∑(t_ij)^2 = ∑(t_ij−t'_ij)^2

f−f′=T_inv(t−t′)

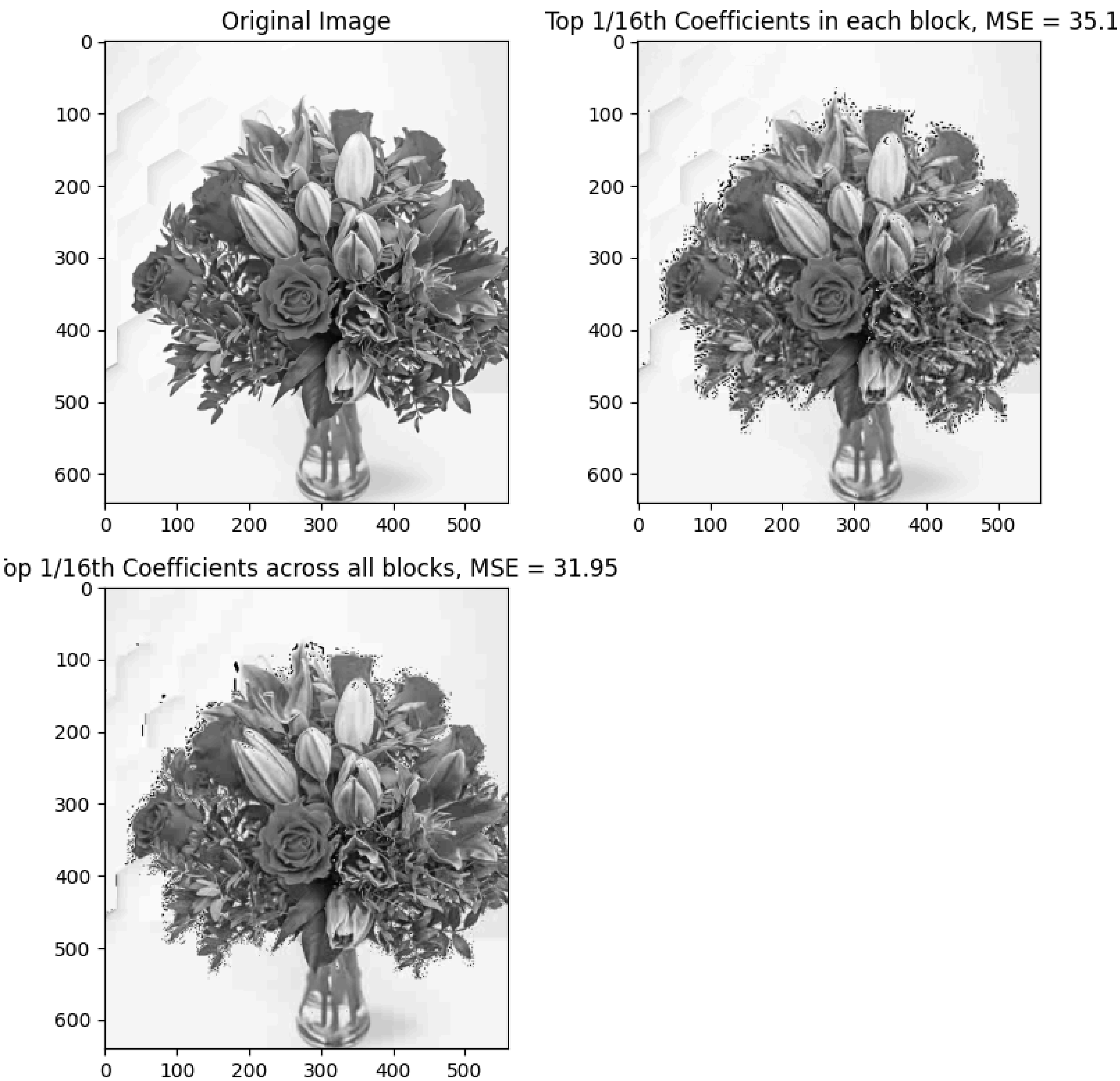
i∑(f_ij−f'_ij)^2 = ∑(t_ij−t'_ij)^2 = deleted tij, ∑(t_ij)^2 ...(2)

Therefore, from (1) and (2)

MSE = 1/MN * deleted t_ij, ∑(t_ij)^2

Part D:
Firstly compute the dct of each 16 X 16 block.
Approximation strategies:

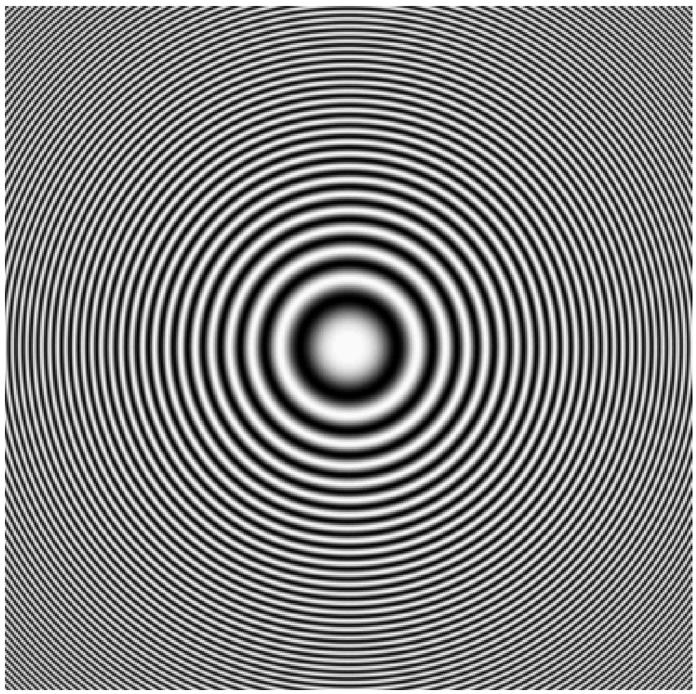
1. **In each block, keeping only the top 1/16 coefficients**
   MSE of top 1/16th for each block : 35.10497767857143
2. **keeping the top 1/16th of all coefficients across all blocks**
   MSE of top 1/16 across all block : 31.952134486607143

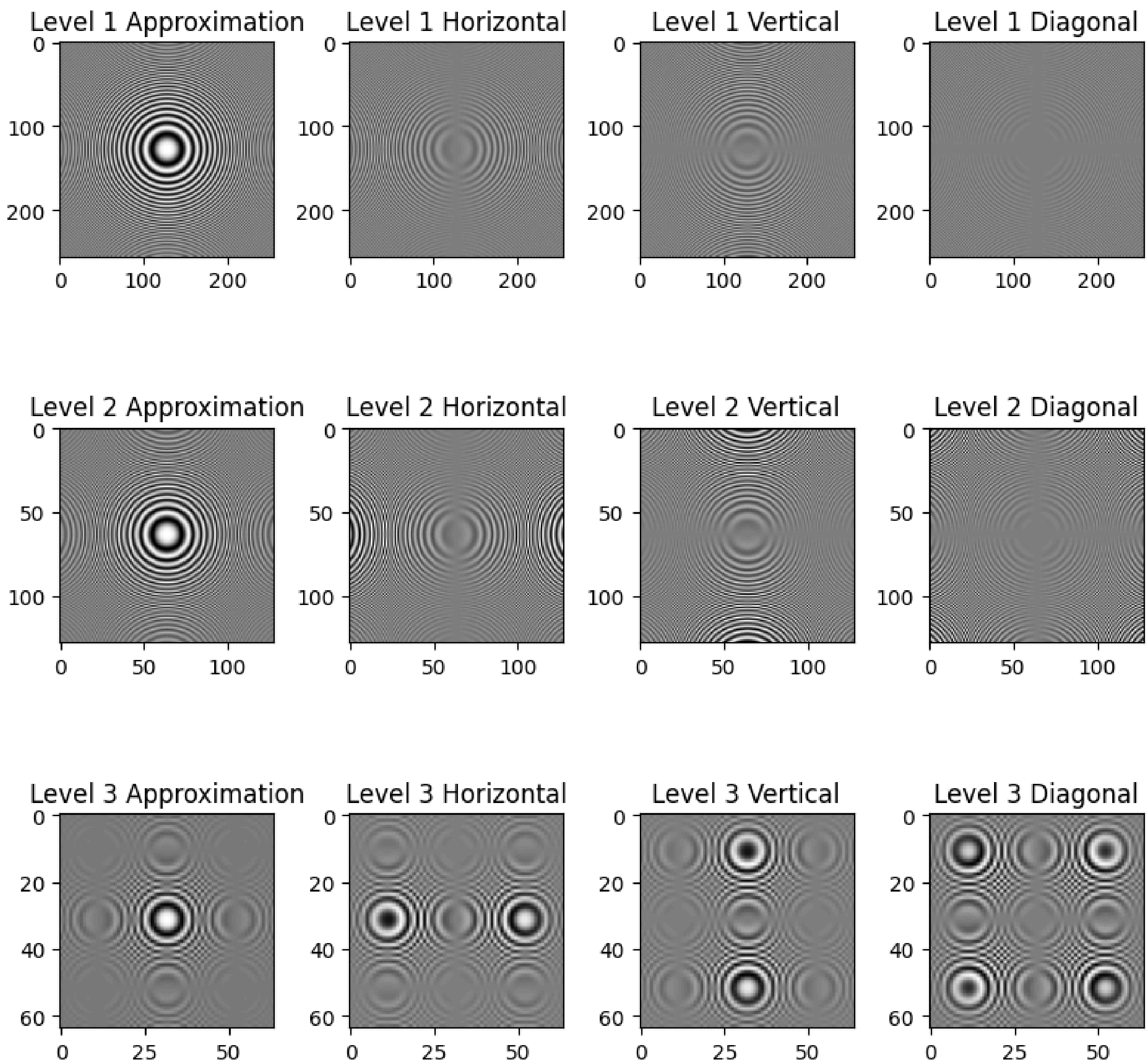


Q2.
Part A:

Zone plate: $z(x, y) = \frac{1}{2}(1 + \cos(x^2 + y^2))$

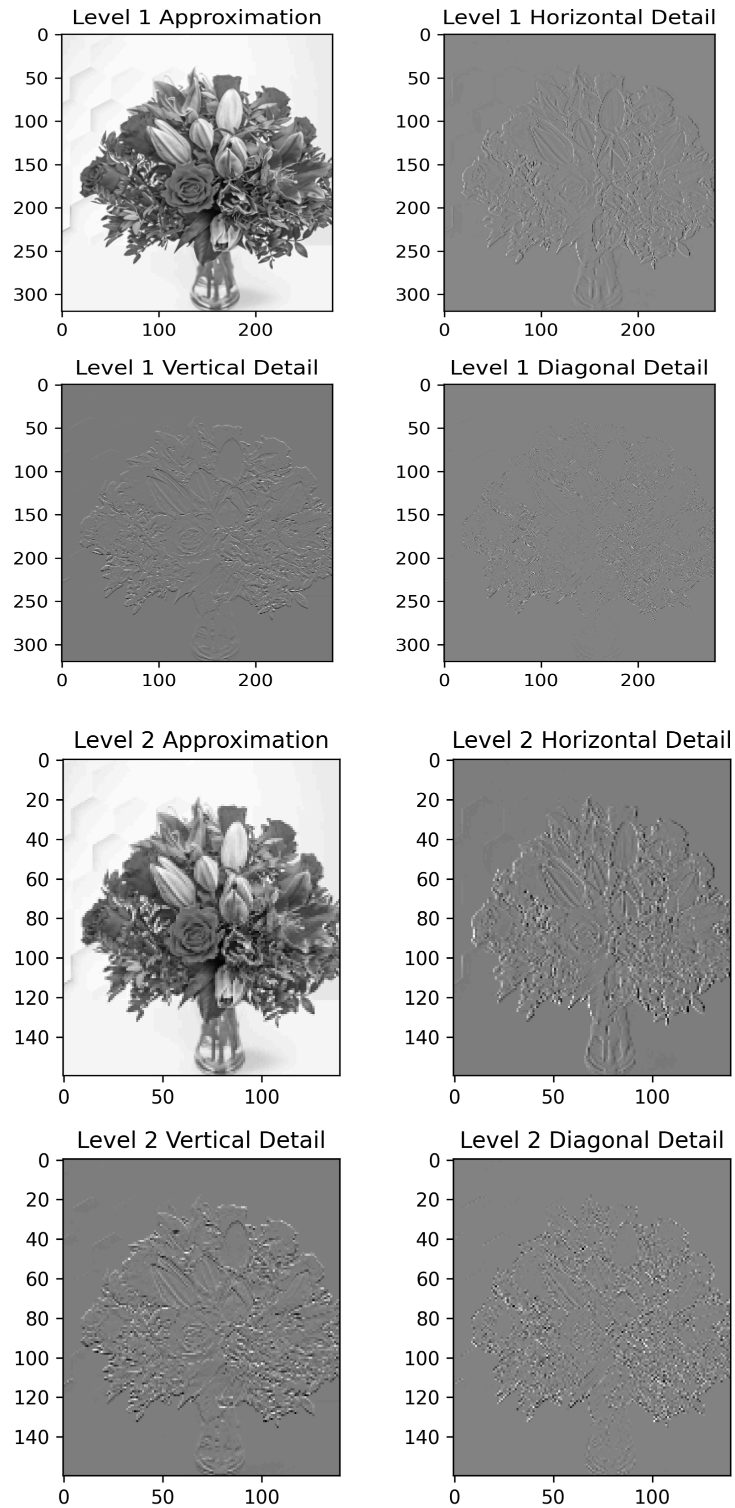
Zone Plate

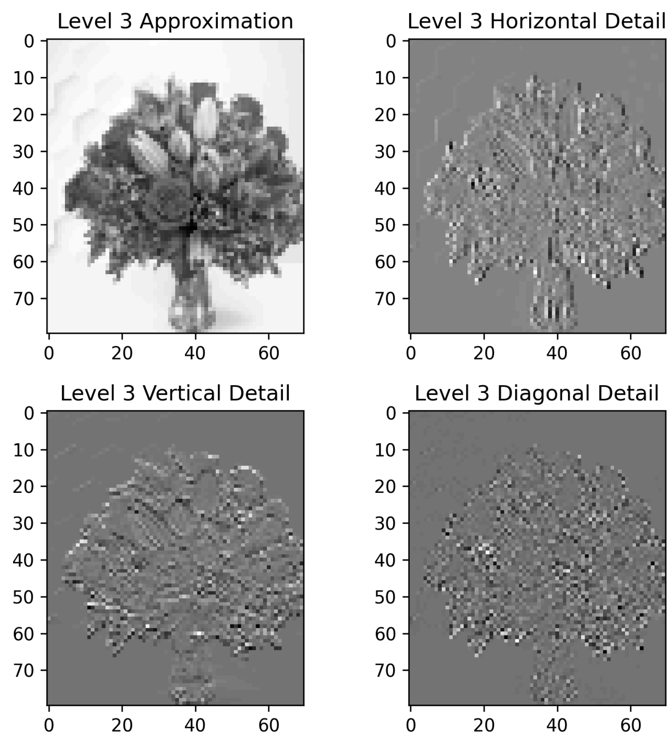




Level 1 Approximation
Level 1 Horizontal
Level 1 Vertical
Level 1 Diagonal
Level 2 Approximation
Level 2 Horizontal
Level 2 Vertical
Level 2 Diagonal
Level 3 Approximation
Level 3 Horizontal
Level 3 Vertical
Level 3 Diagonal

DWT on image f and its corresponding coefficients:





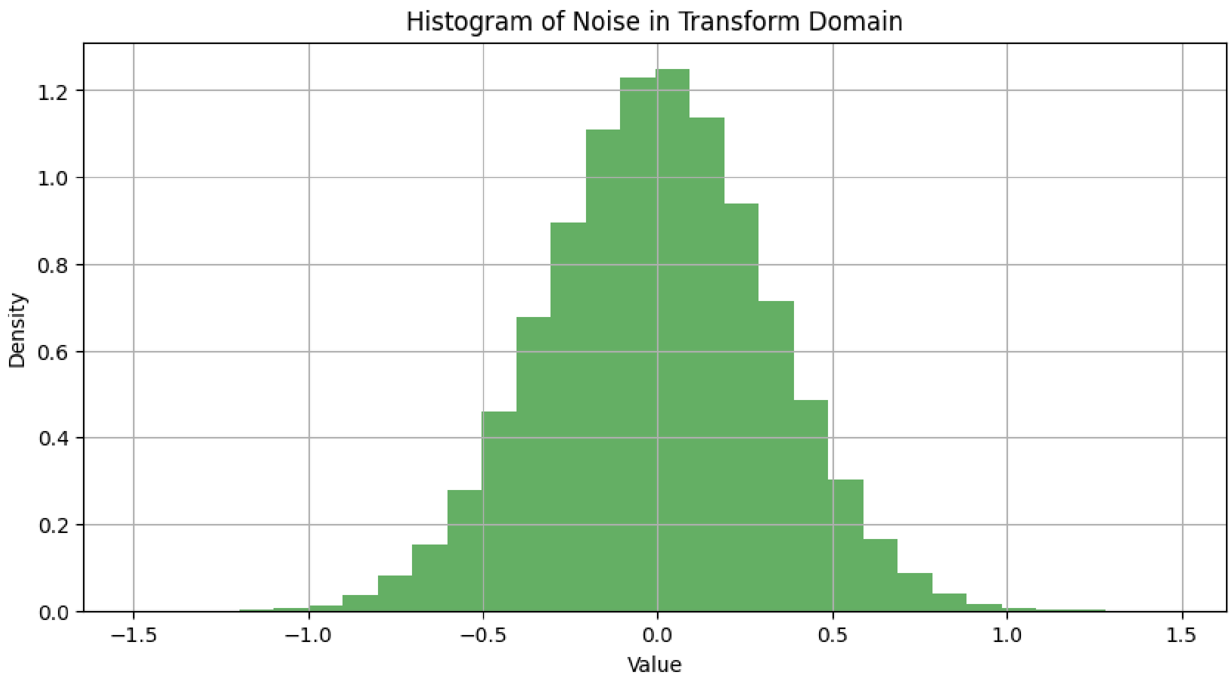
Level 3 Approximation
Level 3 Horizontal Detail
Level 3 Vertical Detail
Level 3 Diagonal Detail

Part B:

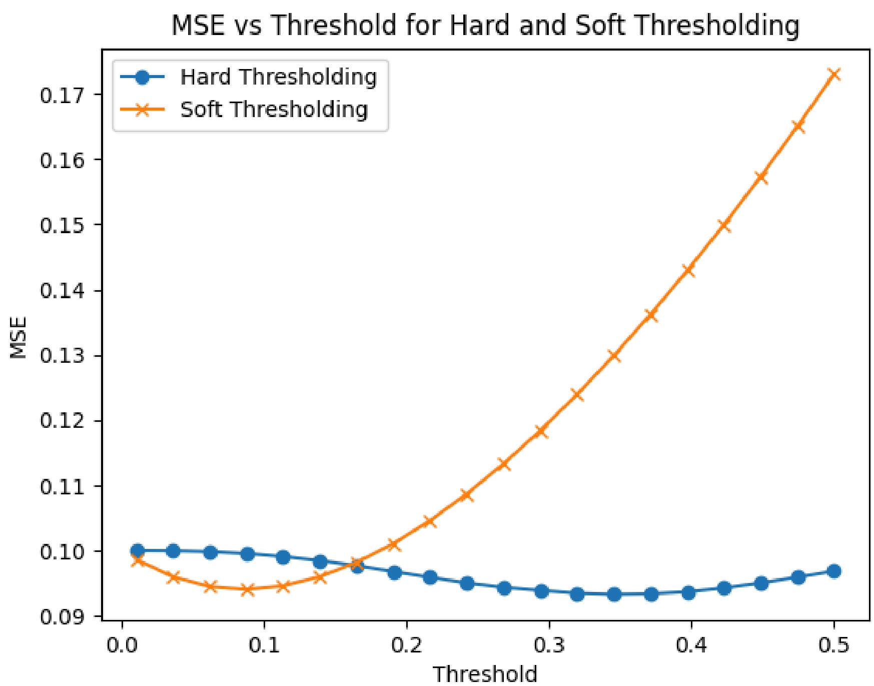
Gaussian Noise:
Mean =0, variance =0.1

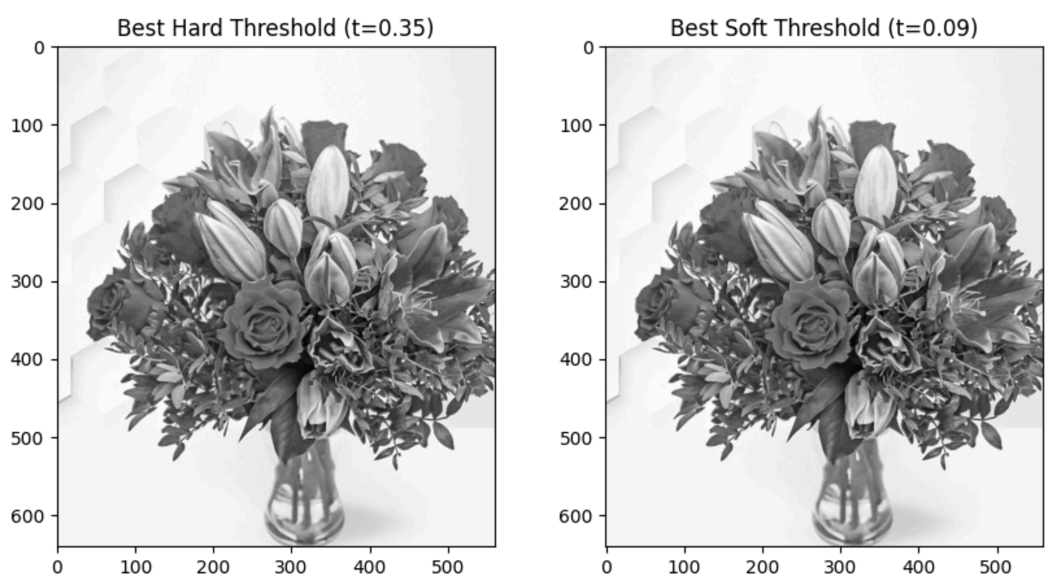
NOISE in transformed domain = dwt{F+ NOISE} - dwt{F}

Noise parameters in transform domain:
Mean of the noise: 0.0006464421706721415
Variance of the noise: 0.10071473076123094
The Noise remains gaussian with the same variance in the transform domain.

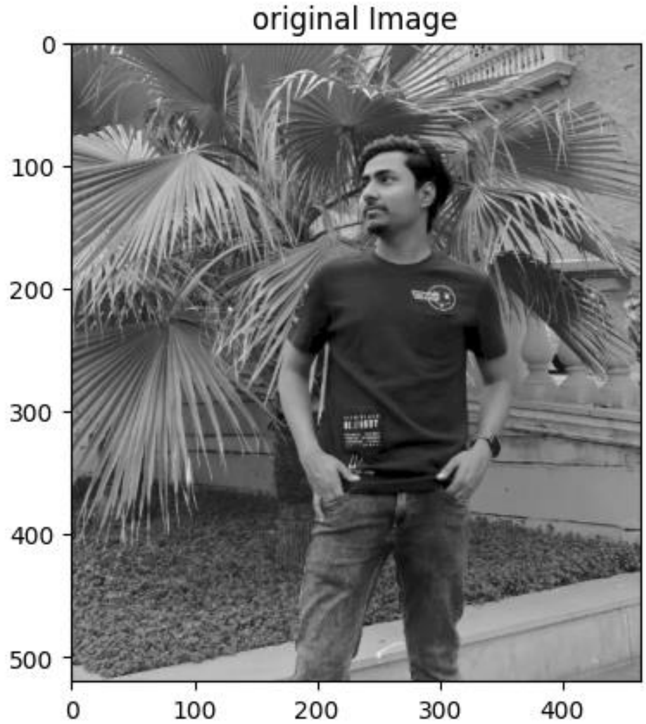


PART C:
Denoising performed by zeroing out detail coefficients of small magnitude.

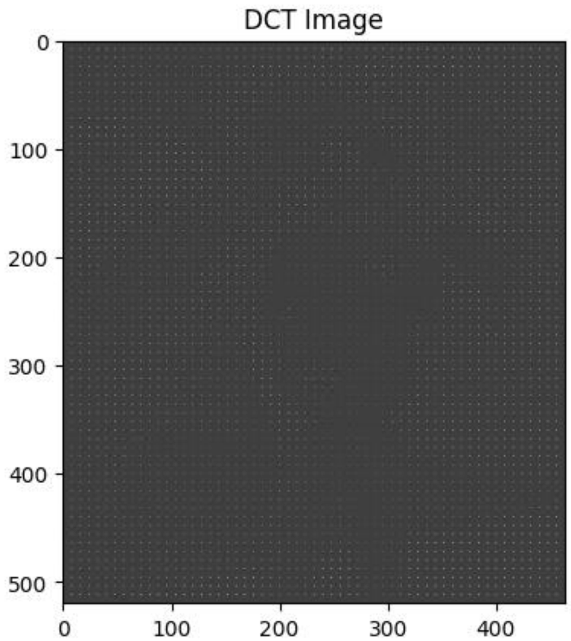


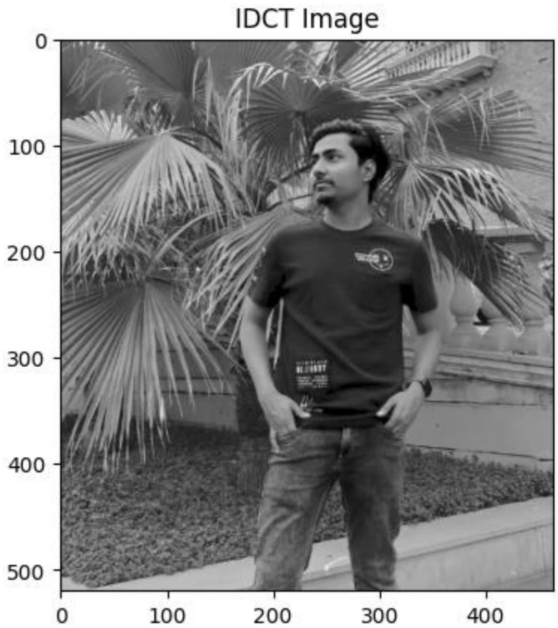


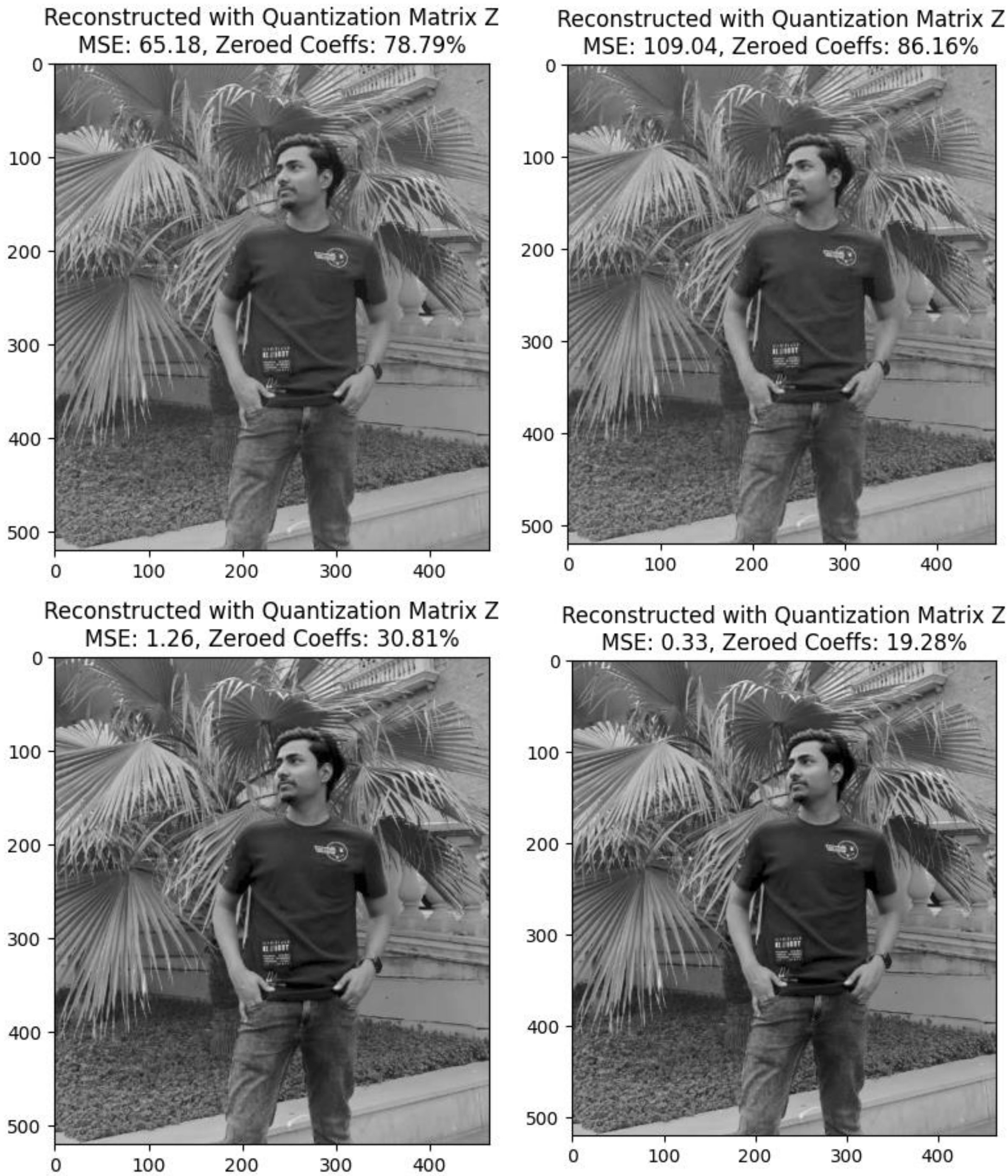
Q3
Part A



Size: (520, 464)



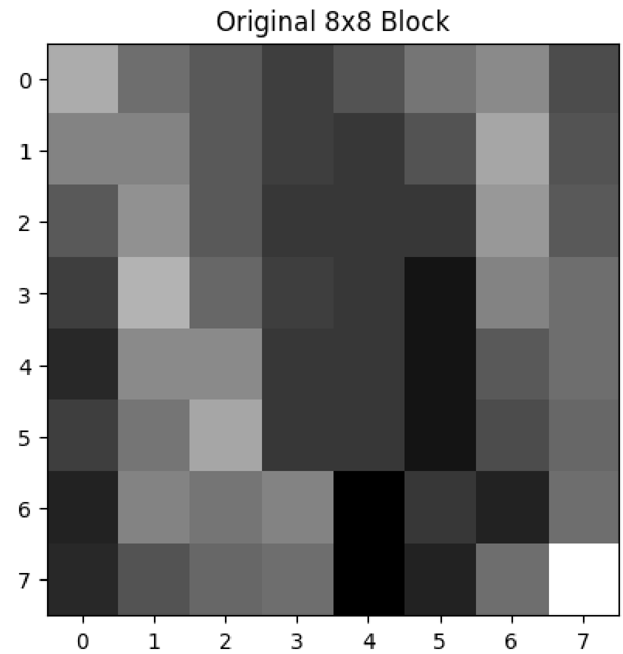




four different choices of Z:

```
quant_matrices = [
    jpeg_quant_matrix,
    jpeg_quant_matrix * 2,
    np.ones((8, 8)) * 4,
    np.ones((8, 8)) * 2
]
```

Part B




```
Quantized Block:
[[52.  0.  2. -1. -1. -0. -0.  0.]
 [ 1.  0.  0.  1. -0.  0. -0.  0.]
 [ 0. -1.  0.  0.  0.  0.  0. -0.]
 [-0.  1. -0.  0. -0. -0.  0.  0.]
 [ 0. -0.  0. -0.  0. -0. -0.  0.]
 [-0.  0. -0.  0. -0. -0. -0. -0.]
 [ 0. -0.  0.  0.  0.  0.  0.  0.]
 [-0.  0. -0.  0. -0. -0.  0. -0.]]
```

```
Reconstructed Quantized Block (8x8):
[[52.  0.  2. -1. -1.  0.  0.  0.]
 [ 1.  0.  0.  1.  0.  0.  0.  0.]
 [ 0. -1.  0.  0.  0.  0.  0.  0.]
 [ 0.  1.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  0.]]
```

Compression Ratio: 7.11

Part C:

Entropy = 5.322022440750344

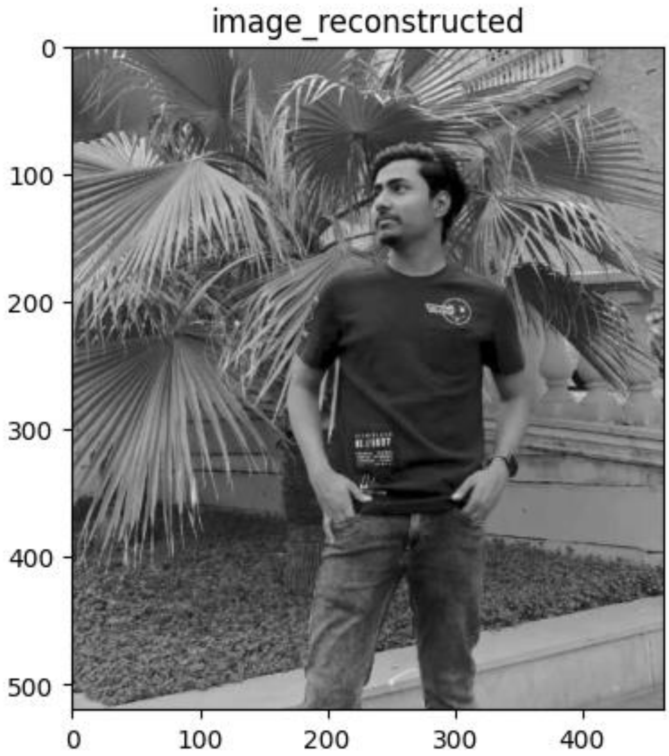
Average Huffman Code Length = 5.3470720966626635

Part D:

Entropy = 5.250353436975073

Average Huffman Code Length = 5.27453870509881

Part E:



final compression ratio = 6.65928371576329