ECE 634 – Project 2

Gautham Vinod

1. VECTOR QUANTIZATION

1.1. Introduction

Vector quantization involves using the LBG algorithm to form a codebook. This codebook is generated by training on an image or images and using the nearest neighbor condition to decide the boundaries of the partitions and then subsequently setting the quantized value as the centroid of these boundaries. Vector quantization as compared to regular quantization helps in preserving the correlation between blocks of pixels.

The LBG algorithm goes as follows:

- Convert the image into blocks of 4×4 pixels.
- Choose the length of the codebook L which is the number of quantization levels and randomly initialize a codebook of $\mathrm{L} \quad 4 \times 4$ blocks.
- For each block of the image, compute the distortion which is the mean squared error between each block of the codebook and find the codeword with the lowest distortion.
- Update the codebook with the block that has the lowest distortion error.
- Calculate the new distortion after updating the codebook.
- Once the percentage difference between the distortion in the previous iteration and the current iteration is less than a threshold, the training is stopped.

After training, we can quantize an image by:

- For each block in the image, find the codeword that has the lowest distortion.
- The block in that particular block index will be the codeword from the codebook that has the lowest distortion error.
- The output image will therefore contain only values that are present in the codebook.

1.2. Results

1.2.1. $\mathrm{L} = 128$

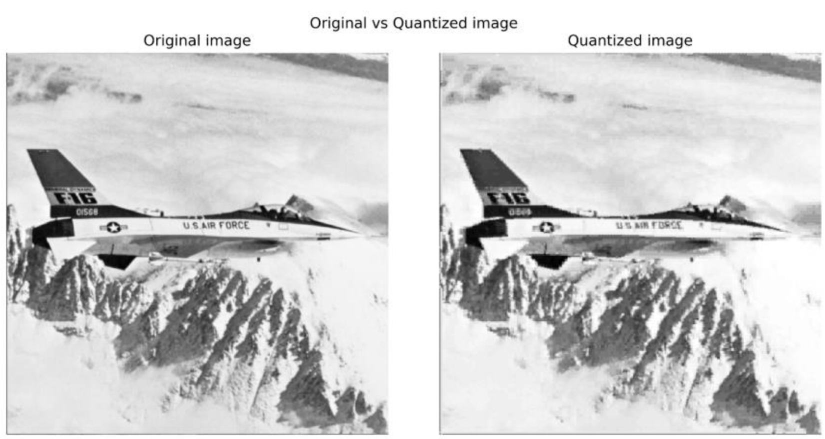

Fig 1: Original vs quantized image of airplane for L= 128. PSNR: 33.63dB

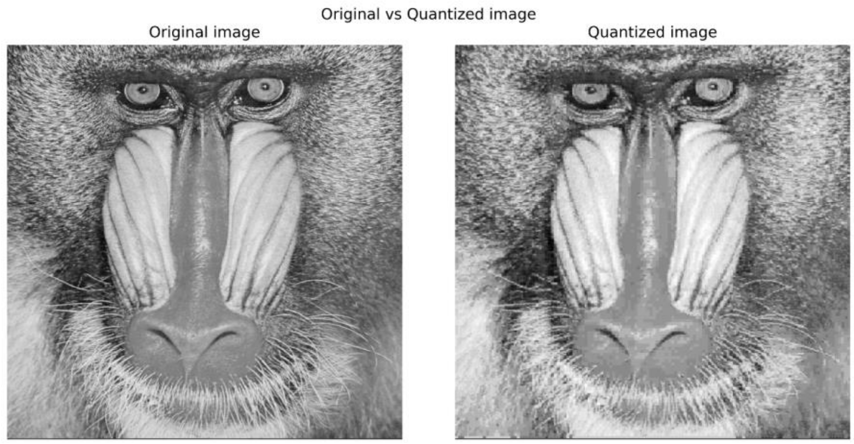

Fig 2: Original vs quantized image of baboon for L= 128. PSNR: 29.5 dB

1.2.2. $\mathrm{L} = 256$

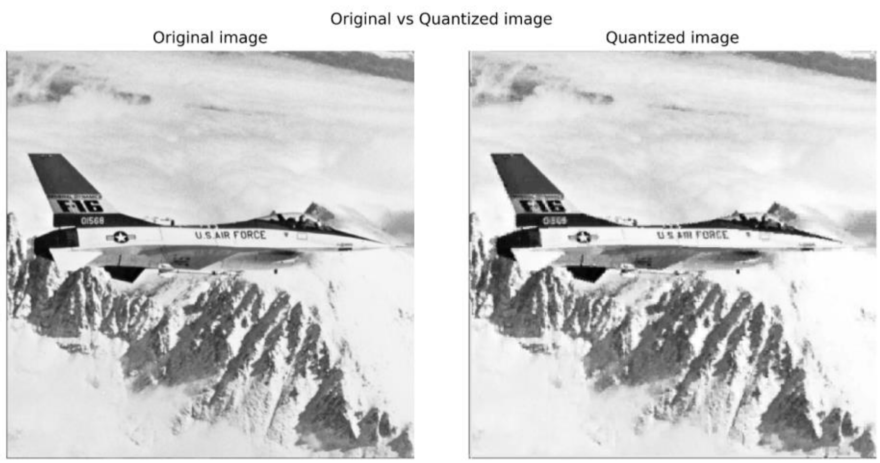

Fig 3: Original vs quantized image of airplane for L= 256. PSNR: 34.11 dB

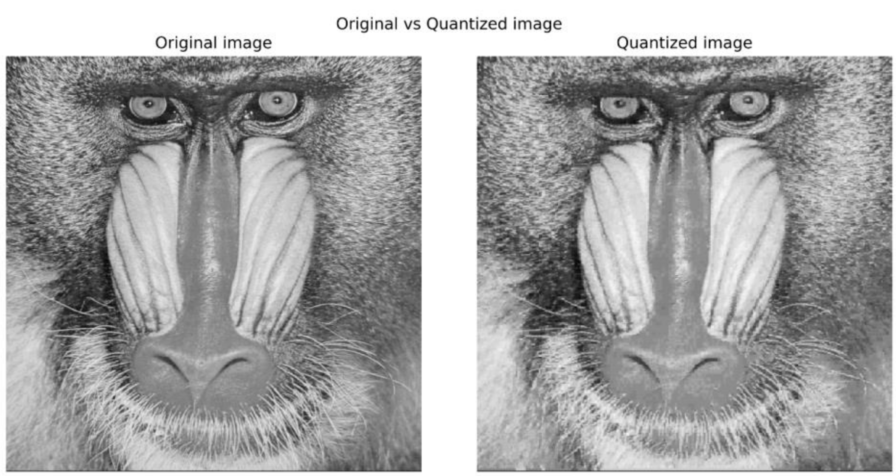

Fig 4: Original vs quantized image of baboon for L= 256. PSNR: 29.67 dB

1.3. Observations

It is clear that a codebook with larger length yields images that are visually more similar to the original image. Although the PSNR values between the 2 cases for both the images do not vary much, the images quantized with $\mathrm{L} = 256$ seem to be of a much higher resolution than that of $\mathrm{L} = 128$.

2. DISCRETE COSINE TRANSFORM

2.1. Introduction

This implementation of DCT involves dividing the image into non-overlapping 8×8 blocks and finding the DCT of each of those blocks. The number of coefficients to keep means that we will first flatten the DCT of each block, sort it in descending order of values and keep the first k values. We look at the reconstruction for different values of $\mathrm{k} = 2, 4, 6, 8, 16, 32, 64$. The remaining values are set to 0.

To reconstruct the images, we find the inverse DCT of each of the vectors that only have k coefficients and then reshape that to an 8×8 block. Each block index of the output image is then substituted with the corresponding inverse DCT block to reconstruct the image. Here the OpenCV functions for DCT and inverse DCT are used. We can look at the MSE between the original image and the reconstructed image to judge the reconstruction quality. The PSNR is another metric that we can use as well but it may not give us much information in terms of understanding the quality of the reconstructed image.

2.2. Results

2.2.1. Airplane Image

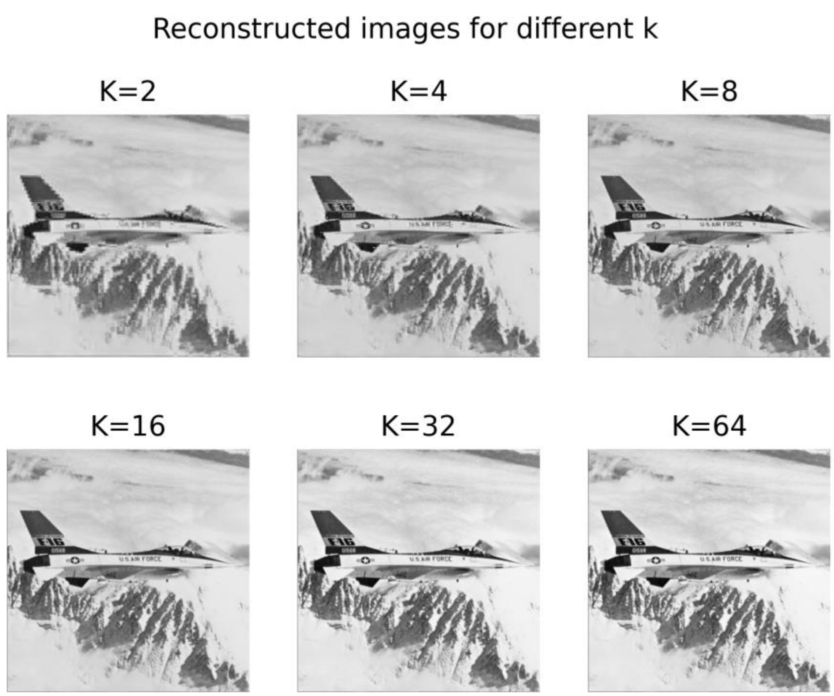

Fig 5: Reconstructed images of airplane using different number of coefficients.

We look at the MSE values of the reconstructed images wr.t. the number of coefficients used.

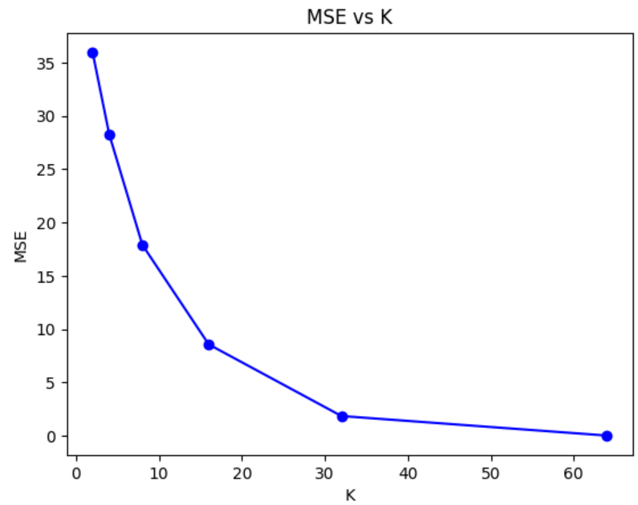

Fig 6: MSE vs k for airplane image reconstruction.

2.2.2. Baboon Image

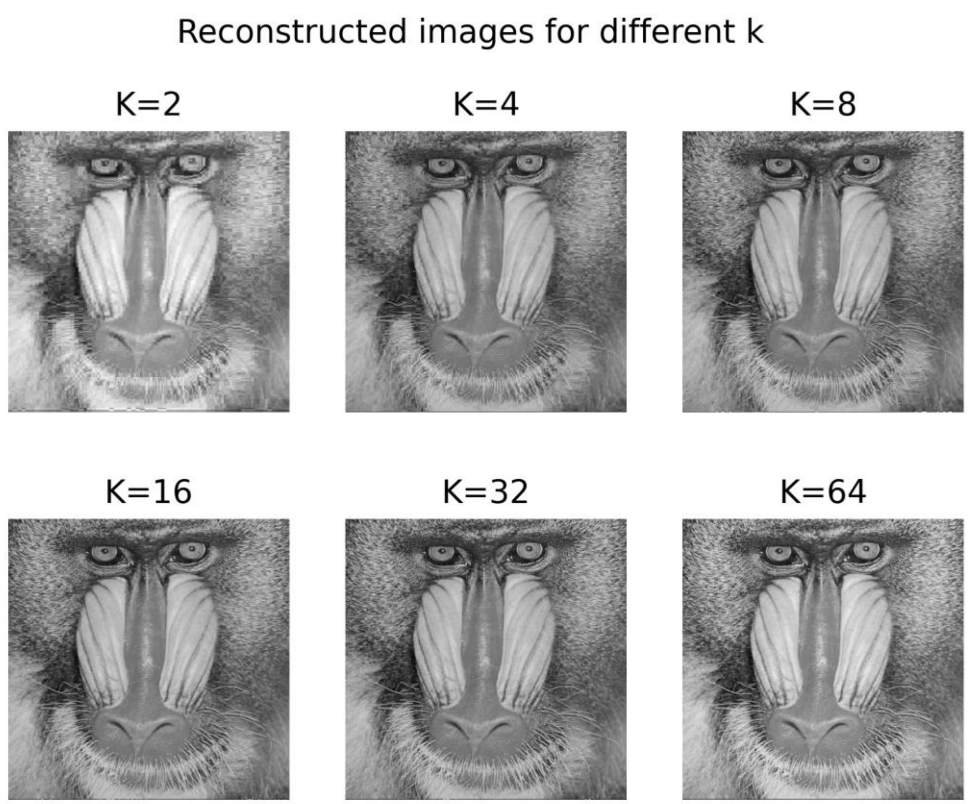

Fig 7: Reconstructed images of airplane using different number of coefficients.

Again, we look at the MSE between the original image and the reconstructed images and plot the MSE w.r.t the number of coefficients retained from the DCT.

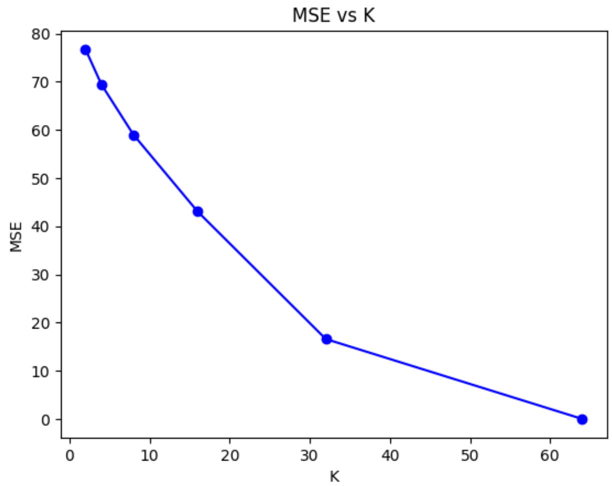

Fig 8: MSE vs k for baboon image reconstruction.

2.3. Observations

It is obvious, from the quality of the reconstructed images, the plot of MSE vs K, from visual inspection of the images, and from a logical point of view that the quality of reconstruction increases when the number of DCT coefficients retained increases. This is because the number of basis vectors that is used in the reconstruction will be larger when the number of coefficients retained increase thereby spanning a bigger space which would provide better reconstruction.

To categorize what can be a considered a "suitable" reconstruction, we can choose to look at purely the MSE between the original and reconstructed image to set a threshold. For example, for the airplane image, we could say that MSE < 15 would mean suitable reconstruction which means we would need to retain $\mathrm{k} = 16$ or higher from the values of k that we chose. From visual inspection, we can indeed see that $\mathrm{k} = 16$ and higher provide very good reconstruction as compared to the original image. However, $\mathrm{k} < 16$ are noticeably blurry and the resolution is lower.

On the other hand, for the baboon image, visually there is not too much of a difference in the reconstructed images from $\mathrm{k} = 8$ onwards. However, the plot shows that the MSE between the reconstructed and the original image for $\mathrm{k} = 8$ is closer to 60. If we were to use our earlier criteria for an MSE cutoff, then we would need $\mathrm{k} \geq 32$. Therefore, choosing a "satisfactory" reconstruction can vary from image to image. Theoretically, the higher the number of coefficients retained, the better, therefore it is a trade-off between the number of coefficients to retain and the reconstruction quality.

3. CONCLUSION

For both tasks, a larger retention increases the quality of the output image. For vector quantization, increasing the length of the codebook increases the quality of the quantized image and reduces the artifacts in the image. In the case of the DCT, retaining more coefficients increases the quality of the reconstructed image. Finding the trade-off between reducing the number of parameters and the quality of the output might vary from image to image.