DIGITAL SIGNAL PROCESSING

(EC-311)

PROJECT REPORT

SUBMITTED BY:

AMMAR AMJAD

SYED MOHAMMAD ALI

QAZI TALHA HAMID

CE-37-A

Table of Contents

Objectives

The aims of the project were:

- Compression of multiple images.
- Transfer of the compressed data to another location as a single file.
- Decompression of the images.

To achieve this, lossy compression is done using Discrete Cosine Transform. Multiple images are compressed and bundled into a single file which can then be transferred to another computer and decompressed to obtain the original images with slight reduction in quality.

Explanation

Compression & Transmission

The compression and transmission is done using steps as follows:

1. Multiple images are selected and individually sent for compression.
2. The first step is to pad the images with zeros so that they can be divided into 8x8 pixel blocks. New dimensions can be found as:

$$\text{newRows} = \text{ceil}(\text{rows} \,/\, 8) * 8;$$
$$\text{newColumns} = \text{ceil}(\text{columns} \,/\, 8) * 8;$$

3. Each image is then divided into 8x8 pixel blocks and on each block Discrete Cosine Transform is applied to obtain the frequency domain equivalent 8x8 block using the formula:

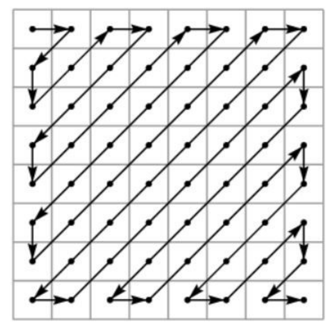
$$F(u,v) = \frac{1}{4}C(u)C(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f(x,y) cos\frac{(2x+1)u\pi}{16} cos\frac{(2y+1)v\pi}{16}$$

$$where\ C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & for\ u,v = 0 \\ 1, & otherwise \end{cases}$$

4. To do lossy compression we can suppress the frequencies that are least contributing to the image and mostly those are the high frequencies. This can be done by dividing the block by a quantization matrix and the JPEG standard has various quantization matrices depending on the quality of the compressed image we want to keep. However, a simple approach is to keep the first four rows and columns and make everything else zeros, like:

$$\mathrm{img}(8{:}{-1}{:}5, :) = 0;$$
$$\mathrm{img}(:, 8{:}{-1}{:}5) = 0;$$

5. The next step is to convert the 8x8 2-D matrix to a 1x64 row vector by picking the elements in a zigzag pattern from the top left so as to have the low frequencies on the left side of the row vector and high frequencies on the right.



6. Since we made high frequencies zero in step 4, this row vector will have a long list of zeros on the right and can be compressed further by using an entropy coding.
7. After converting all the initial 8x8 blocks of the image into these final 1x64 row vectors, they can be concatenated to form a long row vector of size 1xM where M is the product of rows and columns of the image.
8. Similarly, we can concatenate the compressed row vectors of multiple images to form a single row vector that holds compressed data of all the images. This can then be transferred serially to another computer, along with information about rows and columns of every image.

Decompression

The Decompression is done using the inverse of steps used in Compression in reverse order.

1. Using the information about the rows and columns of each image, we can separate the row vectors of every image from the big row vector containing all the images.

2. The row vector of each image can then be further cut into 1x64 size row vectors after applying the inverse of entropy coding done in step 6 of compression.
3. Each of these 1x64 row vectors are converted into an 8x8 matrix by placing them in inverse zigzag pattern.
4. Each block is multiplied by the quantization matrix used in step 4 of compression, or this step can be skipped if high frequencies were directly zeroed out.
5. Inverse Discrete Cosine Transform is applied on each of the 8x8 matrices to obtain equivalent spatial domain 8x8 matrices, using the formula:

$$f(x,y) = \frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7} C(u)C(v)F(u,v)cos\frac{(2x+1)u\pi}{16}cos\frac{(2y+1)v\pi}{16}$$

$$where\ C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & for\ u,v = 0 \\ 1, & otherwise \end{cases}$$

6. All these 8x8 spatial domain matrices are concatenated in the correct order to obtain the image, and similarly recover all the images originally compressed albeit with reduced quality depending on the quantization matrix used.
7. Each image is cropped using information about original rows and columns to remove the padded zeros.

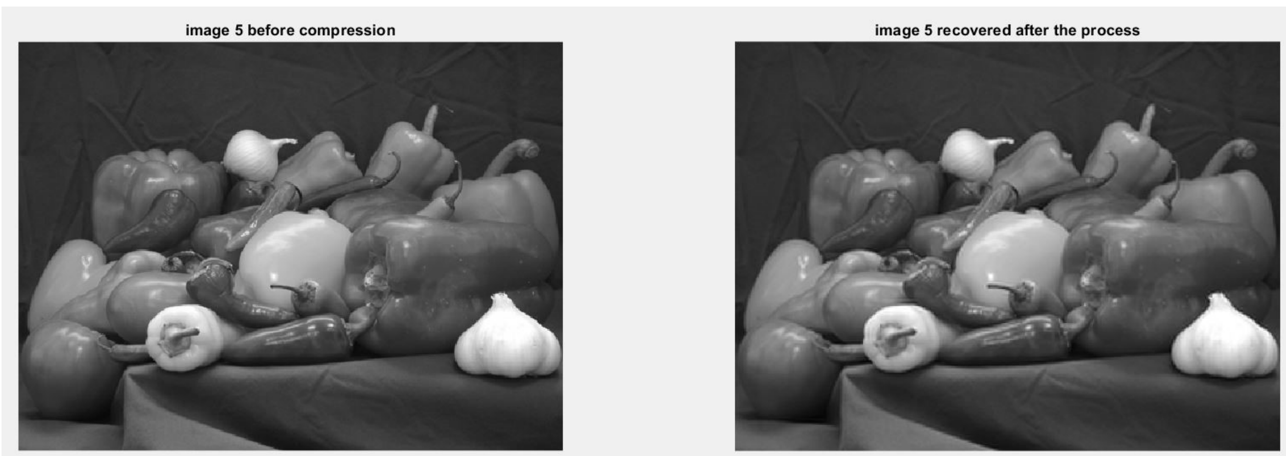
Results:

To test the code, five images were used and sent for compression and converted into a single row vector and then decompressed to get the images back. The results are shown below with the original images on the left and the ones obtained after the process on the right.


image 1 before compression
image 1 recovered after the process


image 2 before compression
image 2 recovered after the process


image 3 before compression
image 3 recovered after the process



This is the comparison of sizes of the images before and after the process:

	Size in KBs of original images	*Size in KBs after the process*
Image 1	11	8
Image 2	15	14
Image 3	39	28
Image 4	18	15
Image 5	19	17
Total	102	82

These results were obtained without Entropy Coding and the size reduction is only due to the lossy compression we performed using Discrete Cosine Transform. Using Entropy Coding will further reduce the size of images.

Conclusion:

Most images contain data that is not contributing significantly to the overall appearance of the image. Usually that data is in the high frequencies which can be removed from the image to decrease the size with minimal impact on quality. Images with a lot of edges however could become blurred if any of the high frequencies representing the edges is removed.

References:

1. Image Compression Using the Discrete Cosine Transform, Andrew B. Watson, NASA Ames Research Center, Mathematica Journal, 4(1), 1994, p. 81-88
2. The JPEG still picture compression standard, G.K. Wallace, Digital Equipment Corp., Maynard, MA, USA, IEEE Feb 1992