CMPE 362
Image Compression Project

Muhammet Ali Topcu - 2020400147

Introduction

Image compression methods are techniques used to reduce the file size of digital images while maintaining an acceptable level of visual quality. There are several image compression methods.

Basically, we can divide image compression methods to two main parts: Lossless compression methods and lossy compression methods. With lossless compression, every bit of data originally in a file remains after it is uncompressed, and all the information is restored. Lossy compression reduces a file by permanently eliminating certain information, especially redundant information.

This project aims to build an application for image compression with 2 different compression methods from literature. One of them is using Discrete Cosine Transform (DCT) which utilizes Fast Fourier Transform (FFT). The other one is using Discrete Wavelet Transform (DWT).

1. Image Compression with DCT

The DCT is a mathematical technique that transforms a signal or image from the spatial domain to the frequency domain. It converts a signal or image into a set of cosine functions with different frequencies.

In image compression, the DCT is commonly applied to blocks of an image. By dividing the image into smaller blocks, such as 8x8 pixels, the DCT is computed for each block independently. The DCT coefficients represent the contribution of different frequencies within each block. The DCT coefficients are then quantized, which involves reducing their precision or resolution. This quantization process introduces a loss of information, as the original coefficient values cannot be perfectly reconstructed from the quantized values. The level of quantization determines the compression ratio and the degree of loss in image quality.

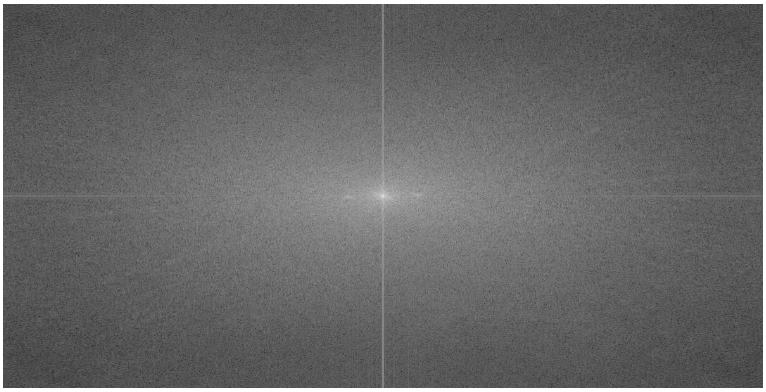
During image reconstruction, the inverse discrete cosine transform (IDCT) is applied to the quantized DCT coefficients to convert them back to the spatial domain. The reconstructed image will have some loss in quality compared to the original due to the quantization step.

The DCT-based image compression techniques have been widely used in various image compression standards, including the popular JPEG (Joint Photographic Experts Group) standard.

DCT (Discrete Cosine Transform) can be efficiently computed using the FFT (Fast Fourier Transform) algorithm in image compression. The DCT is a mathematical transform that converts a finite sequence of data points (such as blocks of pixels in an image) from the spatial domain to the frequency domain. It represents the signal as a sum of cosine functions with different frequencies.

1.2. Implementation Details:

We firstly read the image and convert it to grayscale.
Bt stores transformed grayscale image.
fftshift reorders so that lowest freqs are located at origin and higher frequencies are closer to corners. We take abs value of it since it is a complex number. To display, we take them to log scale and add one to offset 0s.



We show the fourier coefficients in log scale. In the origin, lowest freq components are located. In images, lower freq modes have more energy. We will keep only largest magnitude Fouerier coefficients and we will use them for the inverse Fourier transform.

Then, we will use this Fourier transform for compression. We will manually poll out just the largest 5% coefficients so that we can compress the image. Then, we will zero out the rest and apply inverse Fourier transform and compression is done.

We firstly sort our Fourier coefficients. We turn array of Fourier coefficients into a long vector. Then, we sort from biggest to smallest. Then, we pick a threshold so that we take only top 5%. And we will use this threshold to determine which of those coefficients are larger and keep them and truncate rest of the coefficients.

At line ind=abs(Bt)>threshold, it creates an array of the same size of the original Fourier transform matrix of zeros and ones. It is zero everywhere where the Fourier magnitude is smaller than the threshold and it is one everywhere where the Fourier magnitude is larger than the threshold. ind matrix is basically a mask. It zeros out all the lower coefficients.

Then, we apply inverse Fast Fourier transform. Then lastly, we show the image.

In a for loop, we apply these processes for different compression rates.

In this project, we compressed the image using FFT. In the images, generally Fourier coefficients are low, so we truncated them and compressed the image.

1.3. Examples

Top 99% of the frequency coefficients

Size: 113 KB

Top 10% of the frequency coefficients
Size: 104 KB

Top 5% of the frequency coefficients
Size: 94 KB

Top 0.2% of the frequency coefficients
Size: 42 KB

2. Image Compression with DWT

DWT is a method commonly used for image compression. It is a mathematical technique that analyzes signals and data in both the spatial and frequency domains. The DWT decomposes an image into multiple frequency bands or subbands, allowing for the representation of both low-frequency and high-frequency components.

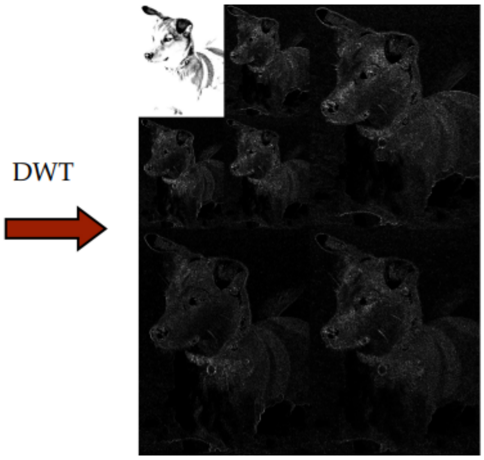
In the context of image compression, the DWT is often used as part of a technique called wavelet-based compression. This technique exploits the ability of the DWT to separate an image into different frequency bands with varying levels of detail. The high-frequency subbands, which contain the fine details and textures of the image, can be quantized and compressed more aggressively since they contribute less to the perceptual quality of the image. The low-frequency subbands, which contain the global structure and smooth areas of the image, can be preserved with higher fidelity.

By quantizing and discarding the less significant coefficients in the high-frequency subbands, and employing various compression schemes for the remaining coefficients, it is possible to achieve compression while retaining a reasonable level of image quality. The compressed image can then be reconstructed by applying the inverse discrete wavelet transform (IDWT) to the compressed coefficients.

Overall, the DWT plays a crucial role in image compression algorithms based on wavelets and has been widely used in various image and video compression standards such as JPEG2000.

2.2. Implementation Details:

We will do Discrete Wavelet Transform and Inverse Wavelet Transform.
We apply wavelet decomposition at line 6. This is 2 level decomposition. db1 is "daubechies 1 wavelet". This is one type of wavelet.



At the left top, we have the course version of the high-resolution image. Firstly, we get a course-grained low resolution version of that image. And each of other levels are corrections to that low resolution version of the image. We take the low-res version of the image and add the other features to it and we get more and more resolution.

Also, all of the refinements are focused on regions where you need more resolution.
In social media or any other platforms where there are images, you may notice that images are loaded firstly with low-res and then, when it completed, it becomes high-res. In these kind of platforms, wavelet decompositions of different levels are loaded gradually.
JPEG2000 was using wavelet.

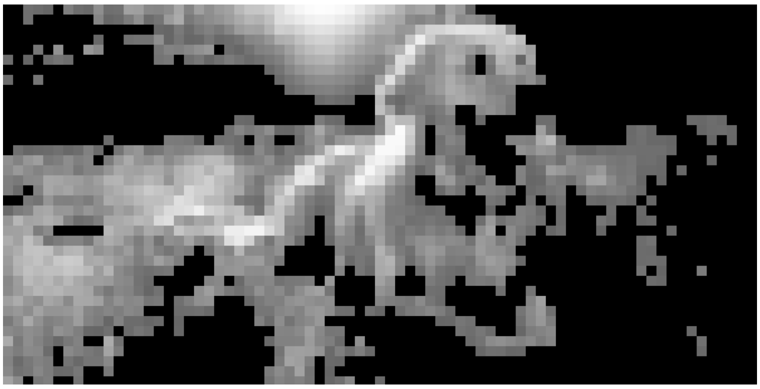
We apply 4 layer wavelet decomposition. Then, we sort these wavelet coefficients and keep first 99%, 10%, 5%, 0.2% of the image and threshold all the rest.

2.3. Examples

Top 99% of the frequency coefficients
Size: 113 KB

Top 10% of the frequency coefficients
Size: 95 KB

Top 5% of the frequency coefficients
Size: 82 KB



Top 0.2% of the frequency coefficients
Size: 12 KB

NOTE: At high frequency components like fur, hair, and edges, wavelets are good at capturing them. But, if we decompose too much like 0.2%, wavelet breaks down as it could be seen. But, FFT was better in that case.

Conclusion

With the heavily increase of digital media and other digital products, images are an indispensable part of everyone's life. For example, in social media applications such as Facebook, Instagram and Twitter heavily depends on images and even videos. So processing them to present users better has an importance. I was very surprised when I read the working principle of DWT image compression. When I was using social media applicatons such as Instagram, I actually realise that firstly image is loaded as low-quality and later, it becomes high-quality. It is very similar to working of DWT in JPEG2000.

References

https://www.youtube.com/watch?v=gGEBUdM0PVc
https://www.youtube.com/watch?v=jclknhNJBrE
https://www.youtube.com/watch?v=KGiV_2i713I
http://databookuw.com/databook.pdf