Video Compression - Homework 2

311551096 錢承

The discrete cosine transform (DCT) is a widely used technique in signal processing and image compression. It has many applications in areas such as digital image processing, audio processing, and video coding. The DCT is based on the idea that any signal can be represented as a sum of cosine functions, each oscillating at a different frequency. This transforms the data from the spatial domain to the frequency domain, where the signal can be analyzed in terms of its frequency components.

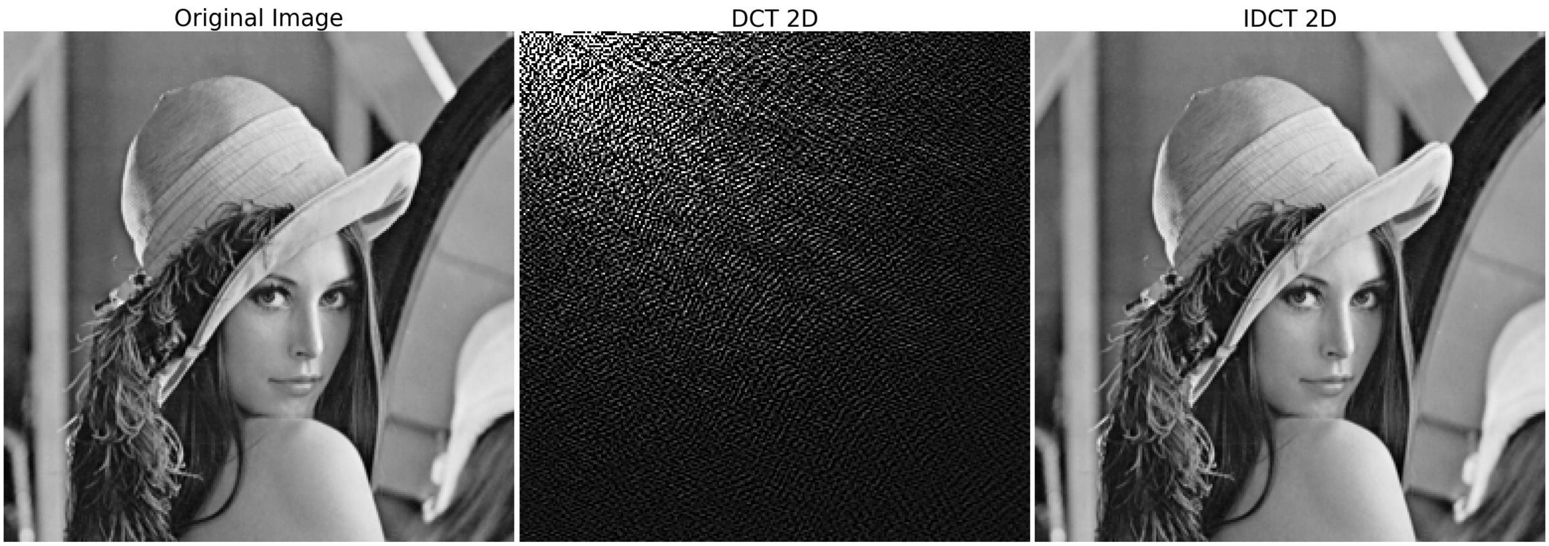
In my implementation of the DCT, I first resize the image to 256x256 to reduce the computing time. I then use two different methods to perform the transform. The first method uses the 2D-DCT formula to convert the images directly. This method takes a long time to compute, approximately 3.78 hours for the 2D-DCT process and 2.56 hours for the 2D-IDCT. However, it produces good results, with a PSNR of 51.094 dB. The following figure is the visualization and the formula of 2D-DCT.

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

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

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

$$0 \le x, y, u, v \le N-1, N^2\text{: frame size}$$



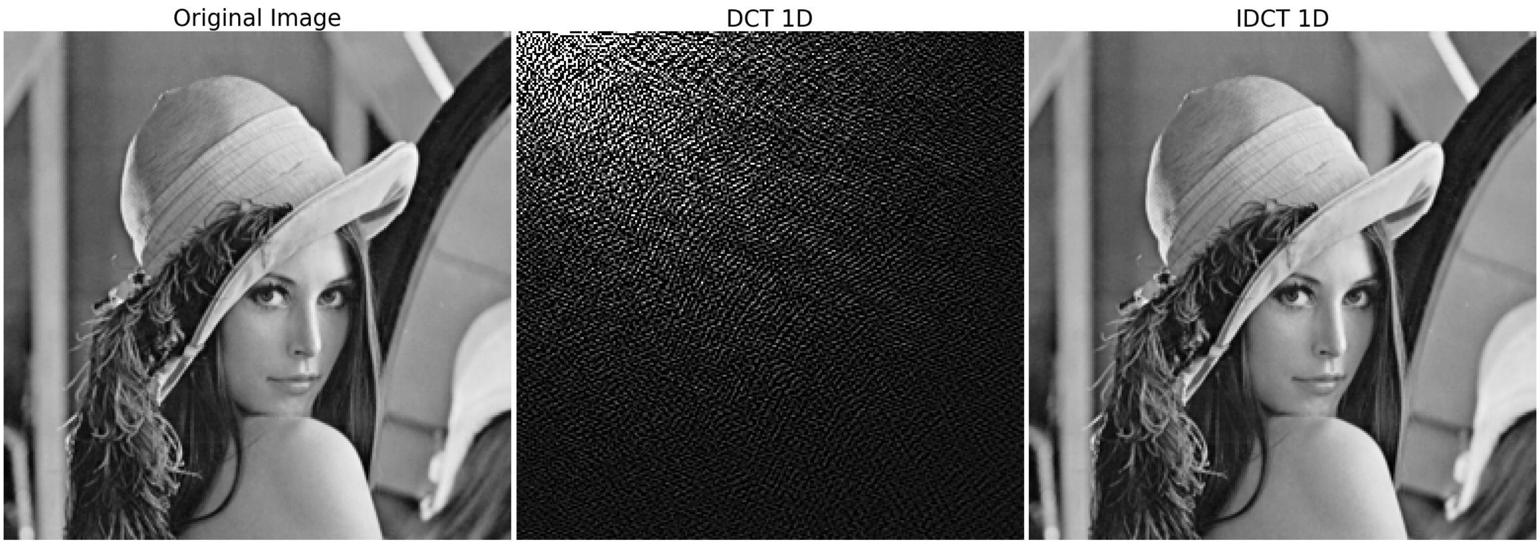
The second method I use is to compute two 1D-DCTs on the image, first row by row and then column by column. This method is much faster, taking only 50.30 seconds for the DCT and 29.82 seconds for the IDCT. Despite its speed, this method produces even better results than the 2D-DCT, with a PSNR of 51.133 dB.The following figure is the visualization and the formula of 1D-DCT.

$$F(u,v) = \sqrt{\frac{2}{N}}C(u)\sum_{x=0}^{N-1} f(x)\cos\frac{(2x+1)u\pi}{2N}$$

$$f(x,y) = \sqrt{\frac{2}{N}}\sum_{u=0}^{N-1} C(u)F(u)\cos\frac{(2x+1)u\pi}{2N}$$

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

$$0 \le x, u \le N-1, N^2\text{: frame size}$$



By comparing the two implementations, it is clear that using two 1D-DCTs is a much faster method than using the 2D-DCT directly. Moreover, this approach produces better results, indicating that it is a more efficient method for image compression. This implementation can be useful for real-time applications, where speed is critical, or for compressing large image datasets where time and computational resources are limited.