Lesson 10: Image Compression

Compression allows us to remove statistical redundancy from our data to encode the same amount of information with fewer bits. ***Lossy compression*** takes this one step further and drops "less important" information to further reduce storage size. We'll be exploring rudimentary lossy compression using the DFT.

Figure 1: Jacob at the Summit of Half Dome

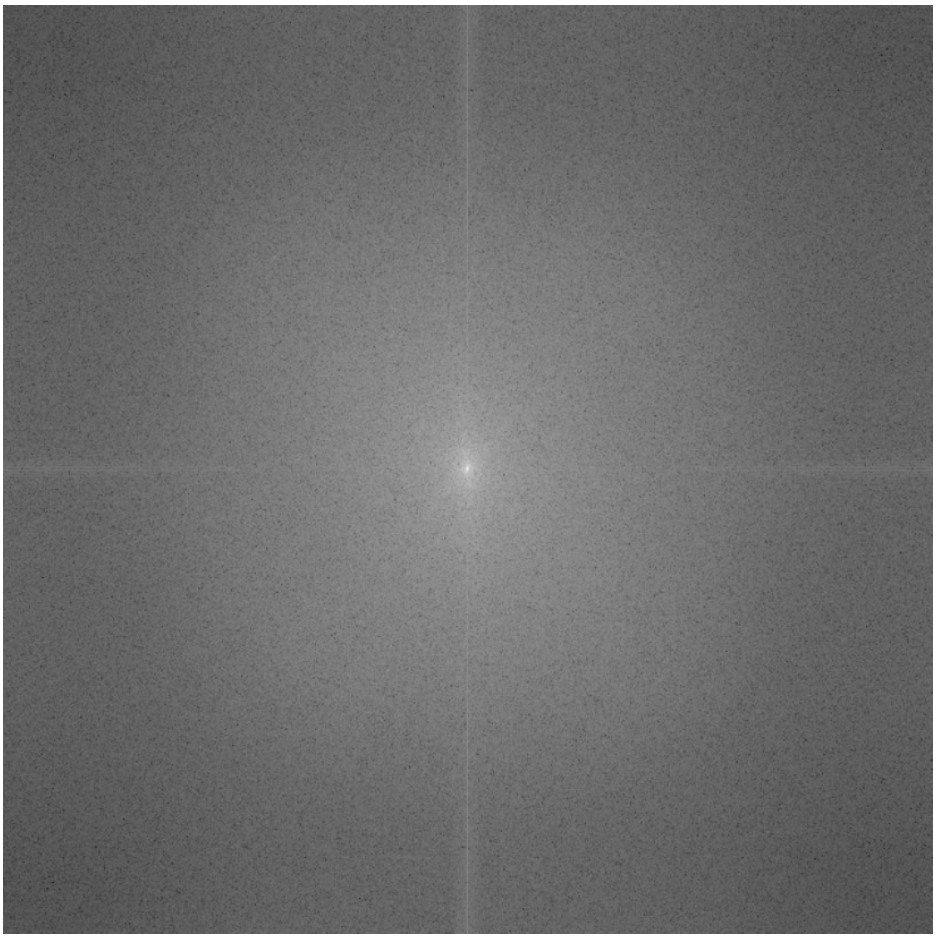
Importing Images

<MINTED>

Here we import the image into MATLAB with <MINTED>, convert it to grayscale with <MINTED> to the simplify analysis, and write it to a file with <MINTED>.

2-D DFT

<MINTED>



Here, we take the 2-D DFT with <MINTED> and use <MINTED> to display its output.

Dropping Samples

Now for the fun: let's keep the "most important" samples by dropping the samples whose magnitude is below a certain threshold:

<MINTED>

Here, we <MINTED> the magnitudes in ascending order and then zero out the samples that are below the threshold. As a last step, we invert the 2-D DFT with <MINTED> and <MINTED>.

Remarkably, this approach offers surprisingly good results, especially when we reduce the resolution of the base image. Although good, we can do much better, but that is far beyond the scope of this course! However, it doesn't take much to appreciate how instrumental compression is to our day-to-day lives.

(a) 00.00%

(b) 90.00%

(c) 99.00%

(d) 99.50%

(e) 99.75%

(f) 99.90%

Figure 2: Dropped Samples

