Assignment 2

Heine Olsson Aabø

October 2018

1 Introduction and Theory

For this assignment I will examine the singular value decomposition and how this can be used to compress an image.

1.1 SVD

The singular value decomposition (SVD) is a matrix factorization $A = USV^T$ where U and V are have orthogonal columns and S is a diagonal matrix with positive real entries. The columns of U are the eigenvectors of AA^T and the columns of V are the eigenvectors of $A^T A$. AA^T and $A^T A$ has the same eigenvalues, which are the square of the diagonal entries of S in descending order.

SVD has various useful applications such as dimensionality reduction and low-rank matrix approximation. The latter can be used to compress an image.

1.2 Compression

The goal of the image compression will be to get a representation of the image that has a smaller number of bytes than the original image. From this representation it should be possible to reconstruct the image, hopefully without any notable differences between the two.

An image of size $m \times n$ has $m \times n \times B$ bytes where B is the number of bytes in the data type representing the image. After the SVD the image is represented by three matrices U, S and V, where U is $m \times m$, S is $m \times n$ and V is $n \times n$. With the low-rank approximation U is $m \times r$, S is $r \times r$ and V is $r \times n$. Since S is a diagonal matrix it can be represented as only the diagonal with size $r \times 1$.

However, for the image to be compressed r has to be of sufficient size, so that

$$mrB + rB + rnB \leqslant mnB \tag{1}$$

Obviously the size of the data type is not important.

$$mr + r + rn \leqslant mn \tag{2}$$

$$r(m + n + 1) \leqslant mn \tag{3}$$

So the maximal value of r needed to compress the image is:

$$r \leqslant \frac{mn}{m + n + 1} \tag{4}$$

The compression ratio is given as

$$compression\ ratio = \frac{uncompressed\ size}{compressed\ size} \tag{5}$$

In terms of n, m and r:

$$compression\ ratio = \frac{mn}{r(m+n+1)} \tag{6}$$

Smaller values of r will therefore give a better compression ratio, but by removing more singular values the difference from the original image will be bigger. It is desirable to find the lowest value of r which gives no notable differences from the original image.

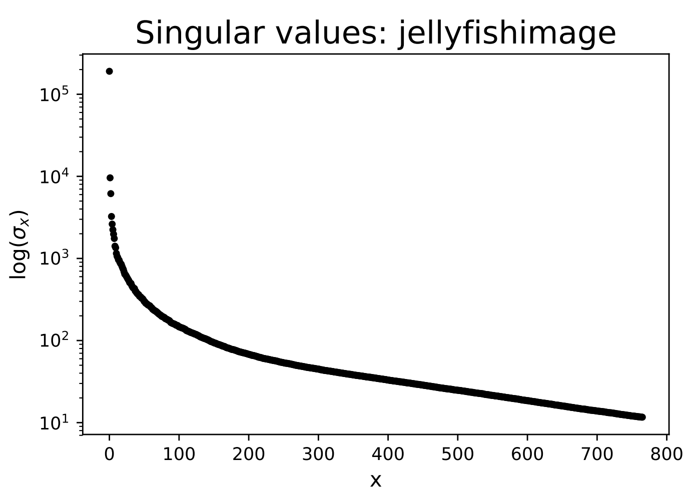
2 Implementation

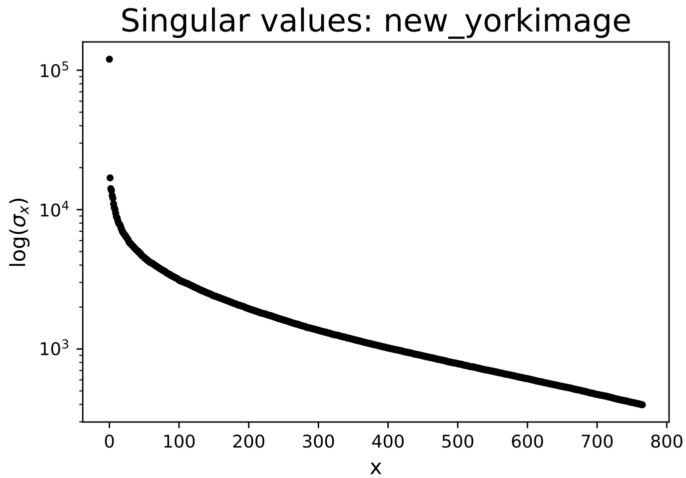
The code can be seen at the end of this report. Here, the uncompressed images are the grayscale images found with the `get()`-function. The pixel values of the images, represented as matrices, are found with `PIL.Image.open`, and then converted to grayscale, where the pixels are only represented with one matrix instead of three. The SVD is done using `numpy.linalg.svd`, and the image is reconstructed using matrix multiplication.

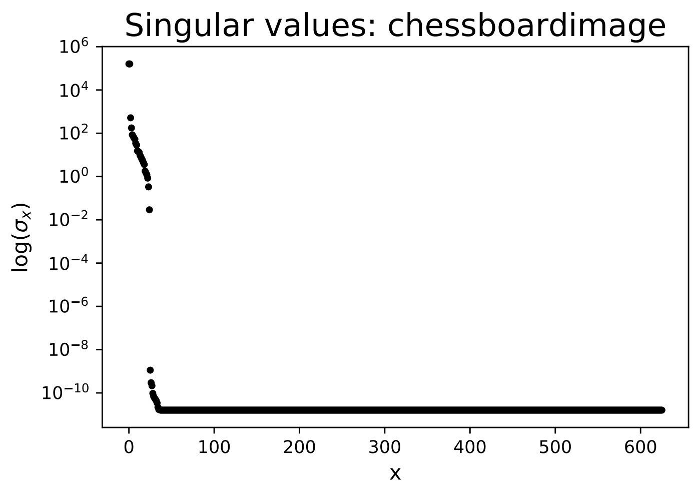
The maxumal r value can be found analytically with eq. (4). To be sure whether lower values of r gives a sufficient reconstruction of the image, a minimal value has to be found by testing different values and consider the difference from the original image.

3 Results

The three images was: of a jellyfish, of a chessboard and of new york. The singular values of the images are plotted below.





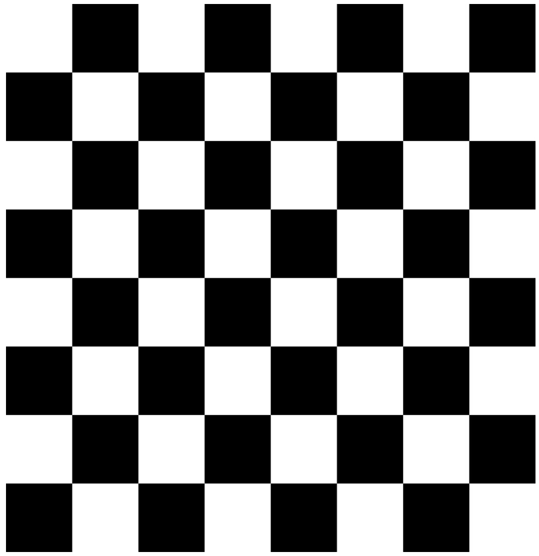


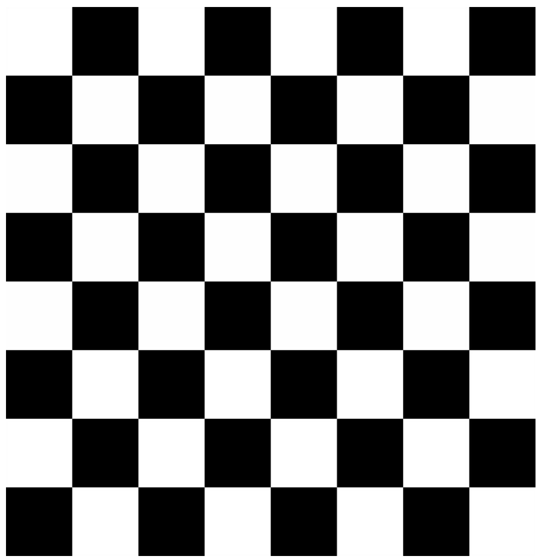
The r values of the images with compression ratios for the minimal r, was found as:

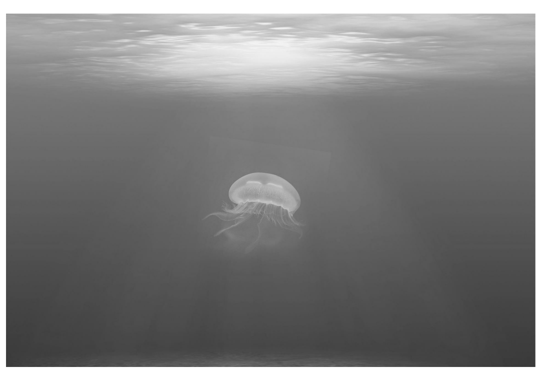
Image	min r	max r	
chessboard	2	765	1.0102
jellyfish	95	765	1.0064
new york	400	625	39.2850

Below is the original grayscale images shown with the images reconstructed from the minimal r values:

(I had some troubles with the captions, so the top image is the original, and the bottom is the reconstructed)







4 Conclusion

The minimal values show what types of images that are easier to compress. The chessboard only needed two singular values to reconstruct the image. This image consist only of vertical and horizontal lines, with very little details. The other images however have much more detail, with the new york image having the most details and thereby needing the most singular values to reconstruct. The images that needed more singular values also got much smaller compression ratios.