CS3414. Extra credit project “*.SVD image format*”

The question

In this assignment you will explore the possibility of using *singular value decomposition* (SVD) for image compression/storage/transmission. The main question you are to answer is this: it is worth it, in your opinion, to invest effort into developing a new image format based on SVD?

Background

Familiarize yourself with the SVD concept and work out a few examples of your own. In the technical part of your report (see below) perform an SVD on a 3x4 matrix with entries (1 2 3) (row 1), (4 5 6) (row 2), etc. Is this a well-conditioned matrix? Read appropriate sections in the C&K textbook. More in-depth discussion of algorithms can be found in the ”Numerical Recipes in C”. Mathematica has routine `SingularValues[]` which may be very useful. Also, Mathematica may prove indispensable for image manipulations. Read chapter 45 in Glynn’s “The beginner’s guide to mathematica.” Be prepared for a (possible) oral discussion about the methods you use.

To Do

You should produce a report with two parts: an **executive summary** and a **technical discussion**. The executive summary should be no more than half a page with one figure (maybe figure 1a and 1b). It is intended for a non-specialist, such as your client. In the summary you give your general conclusion and recommendations. In the technical discussion you give details of your approach. This is intended for specialists, who may want to reproduce your results. This part may be a few pages long. Use figures/pictures. In particular, you should:

a) Take your own picture and transform it into a ”matrix-like” format ready to further manipulations. If working with color proves too difficult, use greyscale, but make comments in your summary about color images and what may be expected. Glynn’s “The beginner’s guide” is a useful source.

b) Perform an SVD on your image, which is now an MxN matrix. You can control the quality of the image by retaining only a certain number of largest singular values σ_n, the quality is then Q = (number of σ_ns kept)/(total number of σ_n). $Q = 1$ means that no data is lost. Try a few different values of Q, perform the inverse SVD transforms, and visually compare the images to the original. What values of Q still give an acceptable quality? How much memory is saved (you need to store the matrices) by using SVD with these Qs?

c) Take a popular "low quality" image format `.GIF`. Convert `yourphoto.JPG` into `yourphoto.GIF`. What is your gain in terms of required memory? Now compare this to `yourphoto.SVD` produced with a Q-value such that `yourphoto.GIF` and `yourphoto.SVD` visually appear to be of the same quality. Supply a few pictures in your report. Do you think `.SVD` is better than `.GIF` for storage purposes?

d) Do you think it is a good idea to use SVD-based compression to send images between cell phones? (Like SprintPCS picture-phone)