Ukrainian Catholic University

Faculty of Applied Sciences

Computer Science Programme

An Application of Linear Algebra to Image Compression

Linear Algebra first interim project report

Authors:
Yulia Hapunovych
Vitalii Papka

06 April 2019

1. The research area of the project

The field we want to cover with our research project is focused on the problem of the images compression. Due to the high demand for transmitting data in the form of images and graphics, fast compression techniques without a loss are required. And linear algebra is directly applied to one of the most important issues in IT: the representation of data by arrays of bits in the most efficient way possible (smaller, faster and cheaper).

Our goal is to make the research on the elegant application of mathematical tools and concepts (in particular from linear algebra and numerical analysis) to the problem of image compression, and illustrate how certain theoretical mathematical results can be effectively used in applications that include transformations in the image representation.

2. The aim of the project

To research, analyze and illustrate the application of Singular-Value Decomposition to image compression technique. To find out for what types of images is this algorithm the best option to use. To evaluate the given algorithm in speed, quality, level of data loss after being implemented using Python and applicable libraries. According to implementation output, to summarize how tools and concepts of linear algebra including matrix manipulation would be useful outside of mathematics, namely to the image compression process.

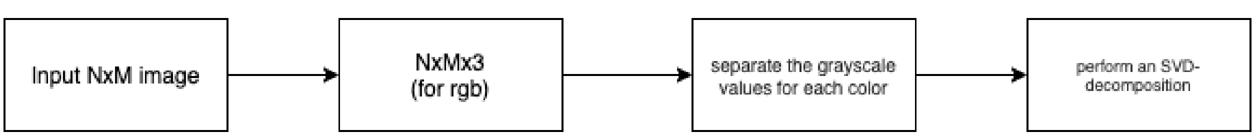
3. Possible approaches and available solutions

Our project undertakes the following approach as a base:

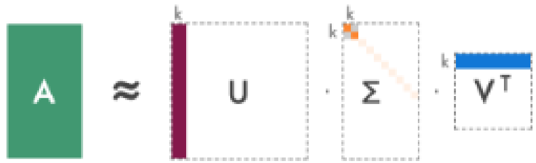
Singular-Value Decomposition (create approximations of an image using the least amount of the terms of the diagonal matrix in the decomposition).Using SVD an image matrix is represented as the product of three matrices U, S, and V where S is a diagonal matrix whose diagonal entries are singular values of matrix A. The image A can also be represented by using less number of singular values, thus, presenting necessary features of an image while compressing it. We will possibly use *numpy.linalg* library's svd function to compute svd of a matrix in python. The svd function returns U, S, V, which we will use further.

4. The pipeline of the implementation

General compression process pipeline:



SVD subprocess pipeline:



5. Testing of implementation

To perform testing will be used sets of images of different size, resolution, extensions, b&w and in colour. For each image, we will try different k-value for rank-k approximation. The implementation of the final algorithm will be evaluated by time, speed, the difference between compressed and original photo (data compression ratio).

6. The plan of future research

- Theoretical research and analyzing of SVD technique (mid of April)
- Providing pseudocode for implementation (end of April)
- Implementation with Python. (mid of May)
- Testing on various image sets (mid of May)
- Visualization of results (end of May)
- Summarising (end of May)

7. Possible challenges to occur

For the current stage, the challenge is to get into the theoretical part of SVD as it is based on the properties of the ordinary diagonalization and uses square roots of the eigenvalues, while this concepts are a bit ahead of our course program.So we need to focus on speeding up our learning of linear algebra to be familiar with all statements used in compression algorithms.