Inchul Pak

Final Project

Linear Algebra and Singular Value Decomposition

As Computer Science and technology change dramatically, we are faced with the situation to deal with enormous big data frequently than before. Especially, computer graphics and image such as picture became more realistic and the quality has improved greatly, and it eventually leads the data to requires a large capacity. Therefore, in order to deal with such a big data, people figured out several methods to compress the data. In this project, we will be discussing how Linear algebra can affect reducing the size of data and compression of images. Furthermore, how singular value decomposition (SVD) technique is extensively used in image compression process resulting in saving computer's capacity.

All the images are made up of millions of pixels by matrix form. An image of resolution m x n is represented as a matrix of values. For example, if we have a 16-megapixel gray-scale image it means 4000 x 4000 pixels (matrix). Every each pixel has a different level of black and white color, given by integer number between 0 to 255. Number 0 representing black color and 255 representing white color.

Singular Value Decomposition (SVD)

SVD method provides stable compression and a clear indication of linear algebra as an image compression tool. Image compression uses a low-rank approximation on the single value decomposition. The SVD formula of the m x n matrix is as follows.

$$A = U\sum V^T$$

U: m x m Orthogonal matrix ($AA^T = U(\sum\sum^T)U^T$)

V: n x n Orthogonal matrix ($A^T A = V(\sum^T\sum)V^T$)

$\sum$: m x n Rectangular diagonal matrix

In this case, the eigenvalues of AA^T and $A^T A$ are both non-negative and non-zero eigenvalues are equal. The eigenvalues must be greater than or equal to 0 so that square root can be implanted and if they are the same, they can be expressed as matrix $\sum$.

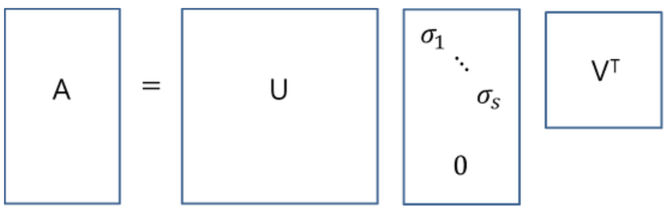
Using the SVD method, it can write an n x n invertible matrix A as:

$$A = U\sum V^T = (U_1, U_2, \ldots\ldots U_n) \begin{bmatrix} \sigma 1 & 0 & 0 \\ 0 & \sigma 2 & 0 \\ 0 & 0 & \sigma n \end{bmatrix} \begin{pmatrix} V1^T \\ V2^T \\ . \\ . \\ Vn^T \end{pmatrix}$$

$$= U_1\, \sigma_1 V_1^T + U_2\, \sigma_2 V_2^T + \ldots + U_n\, \sigma_n V_n^T$$

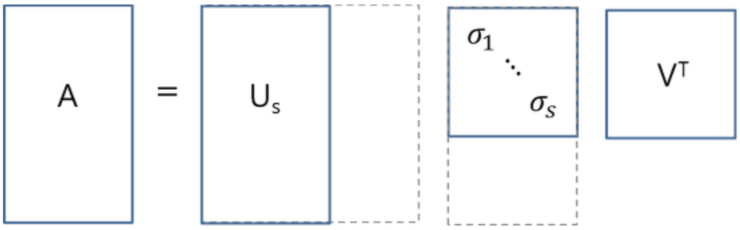
Reduced SVD and matrix approximation

The decomposition of a matrix m x n A into SVD as shown below is called full SVD (m > n).

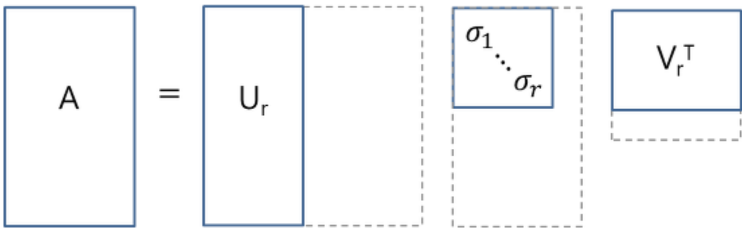


However, it is rare to use full SVD currently and it is common to use reduced SVD as shown in the figure below (assuming r = non-zero, s = n singular values, t < r).

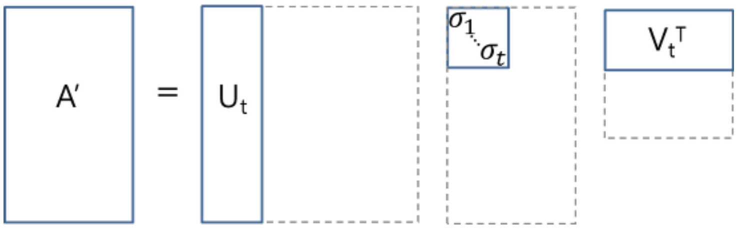
Thin SVD:



Compact SVD:



Truncated SVD:



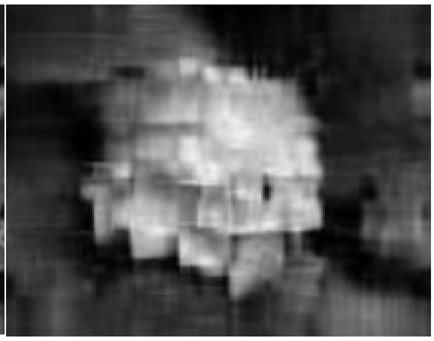
In thin SVD, it removes the part consisting of zeros from the diagonal part in ∑ and removes the corresponding column vector in U. Compact SVD eliminates all the singular values as well as the non – diagonal elements. Both two methods can be easily confirmed that the calculated matrix A is the same as the original matrix A. However, in the case of truncated SVD, a singular value(non-zero) is also removed and the original A is not preserved but an approximate matrix A' comes out. The truncated SVD matrix A' is a rank t matrix minimizing the matrix ||A-A'|| and it can be used for data compression.

Data Compression

As an example of data compression, we can compress the following 768 x 1024 image into SVD.

First, set 768 x 1024 matrix A having pixel values of the image as element values and get an approximate matrix A' by using truncated SVD method. Through the method, the image (approximate matrix A') has a different compression ratio depending on the value of t.

<Output for t = 70>



<Output for t = 10>

To represent the original image, it needs 768 * 1024 = 786432 memory. When t = 10, the compression rate is 17930 / 786432 * 100 = 2.28% since 768*10(U) + 10(∑) + 1024*10(V) = 17930. As we can see the image quality is not good, but it indicates that data approximation A' through truncated SVD is catching the original data core.

Conclusion

There are several methods to compress data, but I think singular value decomposition (SVD) is the simplest and most reliable method that is utilized in many. It has the benefit of providing a decent compression ratio and provides stable and easier ways to split the image matrix into a set of linearly independent matrices. However, the results are from using a black and white image, and It would be interesting to see a result that how full-color image affect SVD compression method.

References

Mounika, K., Lakshmi, D. and Alekya, K. (2015). *SVD BASED IMAGE COMPRESSION*.

[ebook] International Journal of Engineering Research and General Science, pp.Volume 3, Issue 2. Available at: http://pnrsolution.org/Datacenter/Vol3/Issue2/175 [Accessed 2 May 2018].

Dostert, P. (2009). *An Application of Linear Algebra to Image Compression*. [online]

Math.arizona.edu. Available at: http://math.arizona.edu/~brio/VIGRE/ThursdayTalk.pdf [Accessed 1 May 2018].