Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 9, Number 5 (2013), pp. 513-517

http://www.ripublication.com/gjpam.htm

SVD Image Compression: A Classroom Capsule for First Year Linear Algebra

Dr. Tanvir Prince

Assistant Professor, Department of Mathematics
Hostos Community College, City University of New York
500 Grand Concourse, Bronx, New York 10451
Email: TPRINCE@HOSTOS.CUNY.EDU
Office Tel: (718) 518-6828

Abstract

When mentoring a summer research project in 2013, the author noticed a nice application of Singular Value Decomposition in Image Compression which can be illustrated using Mathematica Software. This can be assigned as a project or as a classroom activity for a first year linear algebra students. As a byproduct, students will also learn some simple use of Mathematica software in linear algebra as well as in image compression and image processing. It is also possible to present this as a classroom lecture after the singular value decomposition technique is taught and this will present a nice application of this technique. All the necessary commands of Mathematica which are needed in this project are also given in the paper.

Key word: "Singular Value Decomposition", Mathematica", "Linear Algebra", "Classroom Activity", "Linear Algebra Project"

Introduction:

Now a days, some of the classical topics of linear algebra (like Jordan normal form) are replaced by some of the recent topics such as "Fast Fourier Transform" and "Singular Value Decomposition". These topics usually taught toward the end of the semester and the instructor usually does not find enough time to go into the detail of these topics. This paper proposed a very simple application of singular value decomposition in image compression. Of course, the topic is not new and some of the recent books in linear algebra cover this topic although usually not in great detail, for example, see Introduction to Linear Algebra by Gilbert Strang (Strang, 2009). The idea of this short paper is to show in a comprehensive way of how the instructor can assign this application of image compression using singular value decomposition with the help of mathematica. The instructor can also demonstrate this in a single lecture (in fact only half of the lecture time is needed) right after the technique of singular value decomposition is taught. Although, some basic knowledge in mathematica software is recommended, all the commands are given in this paper which anybody can simply copy and paste in a mathematica notebook and execute the input. For a quick introduction to Mathematica software, the reader can see (Wolfram Mathematica, 2013) and (Purdue University).

Step by step process:
We start with an image. We can simply drug the image into the mathematica notepad.

a = 

We simply denote this image by “a” for late reference. To separate this color image into the corresponding red, green and blue channels, we use the following commands:

$$\{d, e, f\} = \text{ColorSeparate}[a]$$

Here we name the red, green and blue channels by “d”, “e” and “f”, respectively. This produces the following output:

, ,

For detail introduction of the data behind an image can be read in (Image Compression, 2011), (Seaman, 2006) and (Cooper & Lorenc, 2006). Now the singular value decomposition techniques are applied to each channel. Then we take a certain number of singular values starting from the first one. The more singular values are taken, the more accurate the picture is. Typically a picture has huge dimensions in terms of pixels. The command for mathematica to figure out the dimension of a picture is:

$$\mathrm{ImageDimensions}[a]$$

In this case, it is 1024 by 768. So this particular image has 768 singular values. Let us only take the first 40 singular values for each channel (red, green and blue) and then combine the three channels to produce the approximation of the original picture. This will not be as good as the original picture, but this image will take significantly less hard drive space than the original. The command in mathematica to do this is the following:

```
c1=ImageData[d];c2=ImageData[e];c3=ImageData[f];
{u1, w1, v1}=N[SingularValueDecomposition[c1, 40]];
{u2, w2, v2}=N[SingularValueDecomposition[c2, 40]];
{u3, w3, v3}=N[SingularValueDecomposition[c3, 40]];
b1=Image[u1.w1.Transpose[v1]];
b2=Image[u2.w2.Transpose[v2]];
b3=Image[u3.w3.Transpose[v3]];
ColorCombine[{b1, b2, b3}]
```

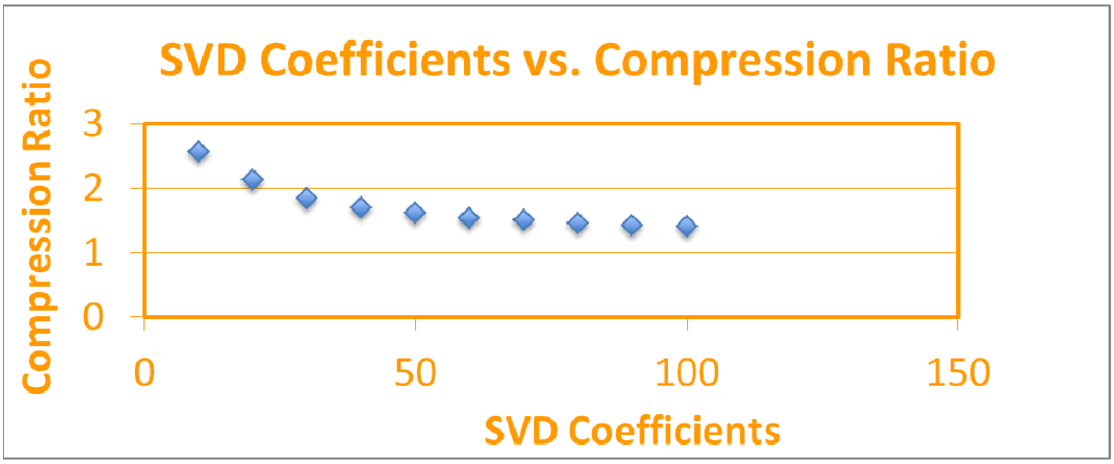
The number “40” appears in the second, third and fourth line of the commands (toward the right) represents the fact that we are taking first 40 singular values from the red, green and blue channels. This number can be change by the reader as he or she pleases.

The output is the following picture:

You may not be able to see any significant difference between this one and the original. This compression only takes the first 40 out of 768 singular values. Thus the compression only takes about 5% of all the singular values and yet it produces a very accurate depiction of the original image. We list below a few more compression with increasing number of singular values (50, 60 and 300 singular values respectively).

Notice that the differences between all these pictures are really hard to catch with naked eyes, only an expert can probably see the difference in quality..

The following graph compares the number of SVD coefficients taken versus the compression ratio. Here the compression ratio is defined to be the quotient of the space taken by the original image and the compressed image.



Conclusion:

Image compression is just one application of singular value decomposition. The instructor can simply show this procedure as a part of the lecture or can assign this as a short project. As a derivative of this project, students will also appreciate the power of mathematica software.

Bibliography

[1] Cooper, I., & Lorenc, C. (2006). *Image Compression Using Singular Value Decomposition*. Retrieved from Retrieved from College of the Redwoods: http://msemac.redwoods.edu/~darnold/math45/laproj/fall2006/iancraig/SVD_paper.pdf

[2] Image Compression. (2011). *Image Compression: How Math Led to the JPEG2000 Standard.* Retrieved from Image Compression: How Math Led to the JPEG2000 Standard. (2011). Retriwww.whydomath.org/node/wavlets/basicjpg.html

[3] Purdue University. (n.d.). *A Brief Introduction to Mathematica*. Retrieved from http://www.cs.purdue.edu/homes/ayg/CS590C/www/mathematica/math.html

[4] Seaman, R. P. (2006). Astronomical Tiled Image Compression: How and Why.. *ASP Conference Series*, 1-4.

[5] Strang, G. (2009). *Introduction to Linear Algebra* (4th ed.). Wellesley-Cambridge Press.

[6] Wolfram Mathematica. (2013). *Hands-on Start to Mathematica*. Retrieved from Wolfram: http://www.wolfram.com/broadcast/screencasts/handsonstart/