IMAGE COMPRESSION Using SVD

Project by,

Nitish S - PES2201800368

Sampreeth Naik - PES2201800179

Supreet Ronad - PES2201800705

INTRODUCTION

- With the advancement in technology, multimedia content of digital information is increasing day by day which mainly comprises of images either pictures or video frames.
- More than 95 million photos are uploaded to Instagram alone every day.

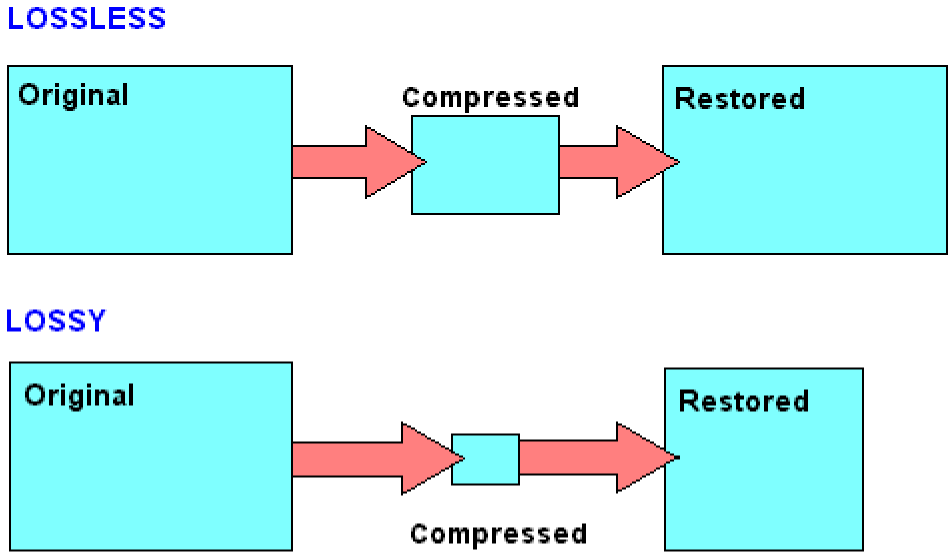
- The storage and transmission of these images is difficult because they need a large amount of space and bandwidth.
- The solution to this problem is to reduce the storage space required for these images which can be done by compressing the image while maintaining acceptable image quality.

What is image Compression?

Image compression means exploiting visual imperceptibility of human eye to encode data with much smaller number of bits for the same amount of information.

LOSSY AND LOSSLESS COMPRESSION

Lossless and lossy compression are terms that describe whether or not, in the compression of a file, all original data can be recovered when the file is uncompressed.



- Many methods are available for compression of still images. But the most widely used image compression technique today is JPEG (Joint Photographic Experts Group) which uses DCT (Discrete Cosine Transform) for compression of images.
- Even though DCT gives high energy compaction as compared to SVD which gives optimal energy compaction, SVD performs better than DCT in case of images having high standard deviation (i.e. higher pixel quality).

What is SVD Method?

- In this method, we take a Matrix A which is the matrix of the input image.
- This matrix is separated into three matrices as shown below

$$\boxed{A_{m*n} = U_{m*m} \cdot S_{m*n} \cdot V^{T}{}_{n*n}}$$

- We apply an algorithm approximation on the three matrices and then combine the new product matrices as like the original image but different in properties.

- The matrix contains the data of the image.
- Any image (RGB) has Red, Green and blue channel.
- We separate the Red Channel, Green channel and Blue Channel from the original image, then apply SVD algorithm on these channels.
- Finally, the output image is the combination of the product channels.
- If we need higher quality we choose a higher value of K(rank).

Advantages Of Svd

- property of energy compaction
- ability to adapt to the local statistical variations of an image
- can be performed on any arbitrary, square, reversible and non reversible matrix of m x n size.
- The psycho visual redundancies in an image are used for compression. Thus an image can be compressed without affecting the image quality.
- The MSE and compression ratio are used as thresholding parameters for reconstruction.

WORKING

1) SVD is a linear matrix transformation used for compressing images. Using SVD an image thought of as a 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 and U and V are orthogonal matrices.
2) The Singular Value Decomposition expresses image data in terms of number of singular values depending upon the dimension of an image. The work is concentrated to reduce the number of singular values required to reconstruct an image.
3) The compressed image requires less storage space as compared to the original image.

Why is SVD used for Image Compression:

WKT ‘Rank of the matrix is equal to the number of non-zero singular values and as the rank of the matrix increases, the number of singular values decreases’.

This property is useful in compressing the image (i.e in terms of matrix data) by eliminating the smaller singular values.

Implementation

Selection of principal components to reproduce an approximated version of the original image. A limit over the number of components chosen must exist. **Important!!!**

Procedure:

If image is thought of as a matrix, then:

Original size of image: m*n ,The overall size increases with that of originally being $m \times n$ to $m \times m + n \times n + n$ ($n<m$).

A= u1.σ1.v1^T + u2.σ2.v2^T…………+ ur.σr.vr^T where r = rank of matrix

A = u1.σ1.v1^T + u2.σ2.v2^T…………+ uk.σk.vk^T   (k<r)

Compression ratio: $(m \times n) / (m \times k + n \times k + k)$ for the k principal components.

For compression to be effective, this ratio must be greater than 1. We can derive an expression for the no. of maximum components k we may select so that the size of the selected components remains less than the original image as shown below..

$$m \times n / k \times (m+n+1) > 1 \quad or \quad k < m \times n / m+n+1$$

This expression is being used for the SVD based compression ratio and the maximum rank permissible for the purpose of image compression.

Thus, by choosing an appropriate value of ‘k’, we can achieve a fairly good amount of compression without much loss in the quality of the image.

Value of k & compression ratio???

1) Choosing the value of ‘k’ i.e the number of Singular values is an important decision for acceptable reconstruction which varies with the application.
2) It is observed that the chosen value of ‘k’ is directly proportional to the quality of the image.

Eg: If the value of ‘k’ is equal to the rank of the image (matrix), then the reconstructed image is close to the original image.

1) Another observation is that lower the compression ratio, higher the quality of image.

RESULTS

Compressed

Original Image

Mode 50

Mode 25

CODE

Modes are how many columns of the matrix **U** you want to use and how many rows of the matrix V^T you wanted to use to calculate your specified **level of Precision.** Mode is similar to K value.

```
function imCompressed=compress(imFullOneChannel, SingularValuesToKeep)
[U, Sigma, V] = svd(imFullOneChannel);
SingularValues = diag(Sigma)
imCompressed = U(:,1:SingularValuesToKeep)*diag(SingularValues(1:SingularValuesToKeep))*V(:,
1:SingularValuesToKeep)'
endfunction

im = imread('C:\Users\DELL\Downloads\whatsapp\LA\a4.jpg');

imFull = double(im);

imCompressed(:, :, 1) = compress(imFull(:, :, 1), 10);
imCompressed(:, :, 2) = compress(imFull(:, :, 2), 10);
imCompressed(:, :, 3) = compress(imFull(:, :, 3), 10);

imCompressedFinal = uint8(imCompressed);
imshow(imCompressedFinal)
```

Without RGB Compression

Original Image

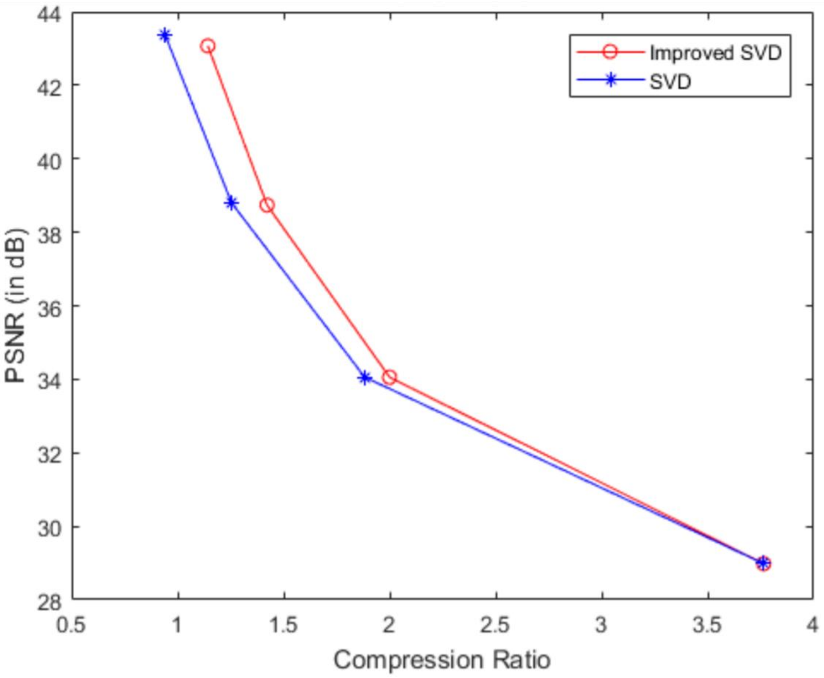
Compressed Image (Mode - 50)

Peak Signal to Noise Ratio or **PSNR** is an important metric for assessing **quality** of the compressed image with respect to the original image.

$$PSNR = 10log_{10}\left(\frac{R^2}{MSE}\right)$$

Where, **R** is the maximum signal value 255 in case of gray level image.

$$Compression\ Ratio = \frac{Size\ of\ original\ image}{Size\ of\ compressed\ image}$$



Structural Similarity Index or **SSIM** is a measure for the distortion with respect to the human visual system.

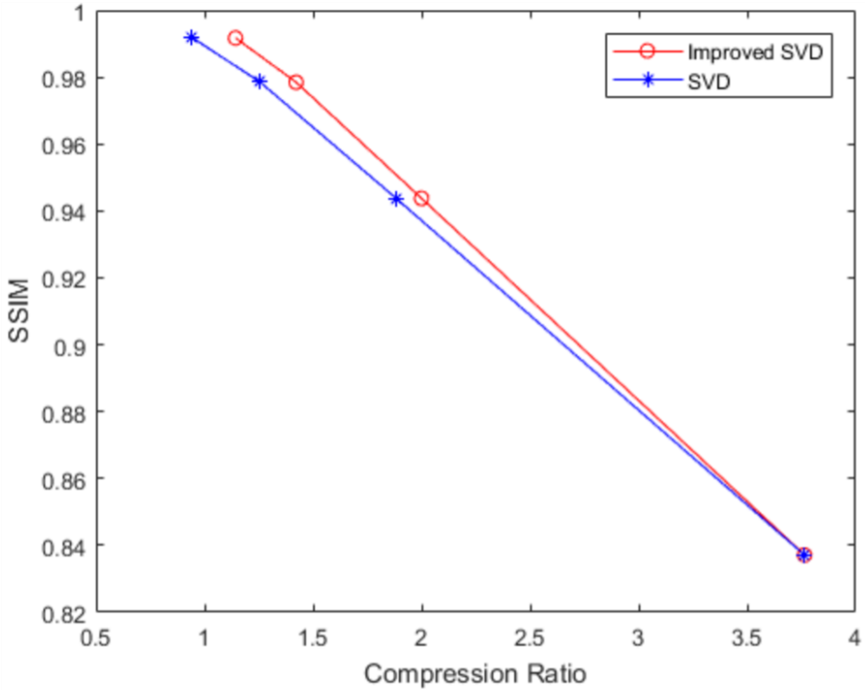
$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x{}^2 + \mu_y{}^2 + c_1)(\sigma_x{}^2 + \sigma_y{}^2 + c_2)}$$

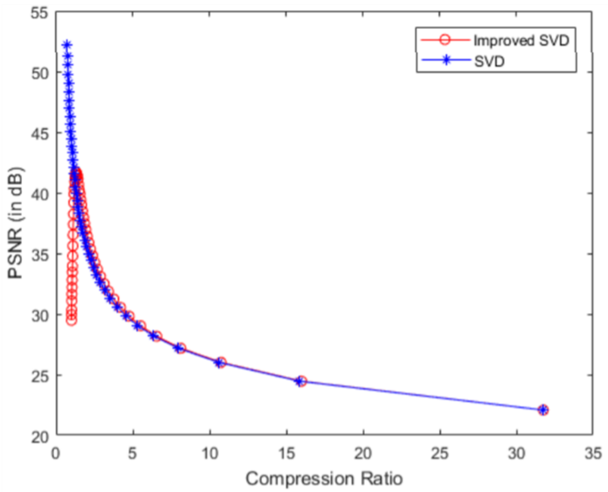
Where,

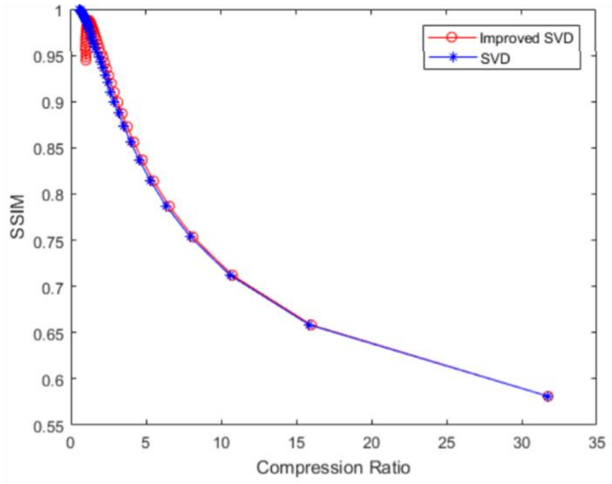
μ_x, μ_y denote the mean value along x and y respectively;

c_1, c_2 denote two variables to stabilize the division when denominator is too small; and

σ_x, σ_x denote the variance along x and y respectively.







Based upon the results, following observations can be made:

- It is very clear by the results that **compression ratio increases** by the new technique while **PSNR and SSIM remain the same.**

- For the block-size 64, the graph shows decrease in the PSNR and SSIM , while theoretically there should be no change. The decrease is due to the precision error in calculating the values upon solving the linear equations.

Applications

This technique is used in applications where a **little compromise on quality** of image is acceptable. This compression not only helps in **saving storage space** but also enables **easy sharing of files**. Some of the main applications are :

- Broadcast Television
- Remote sensing via satellite
- Military communication via radar, sonar
- Teleconferencing
- Computer communications
- Facsimile transmission
- Medical images : in computer tomography
- Magnetic Resonance Imaging(MRI)
- Satellite images, geological surveys, weather maps

Conclusion

Not much work had been done in SVD based image compression, reason being its **high complexity**. But the technique can be further extended to **identifying similar blocks and skipping them** to deliver much more efficient performance. Image compression applications reduce the size of an image file **without causing major degradation to the quality** of the image. The compression **tools are user friendly** and can be used by anyone with minimal knowledge. Thus they result in **efficient utilization of time, memory and bandwidth**.

Thank you