Image Compression using PCA

G.Sai Ram pavan AM.EN.U4AIE20125
Department of Compter Science(Artificial Intelligence)
Amrita Viswa Vidyapeetam
kollam,kerala, India
amenu4aie20125@am.students.amrita.edu

K.Akash Varma AM.EN.U4AIE20141
Department of Compter Science(Artificial Intelligence)
Amrita Viswa Vidyapeetam)
kollam,kerala, India
amenu4aie20141@am.students.amrita.edu

N.Moneesh AM.EN.U4AIE20150
Department of Compter Science(Artificial Intelligence)
Amrita Viswa Vidyapeetam
kollam,kerala, India
amenu4aie20150@am.students.amrita.edu

Sandeep Neemkar AM.EN.U4AIE20163
Department of Compter Science(Artificial Intelligence)
Amrita Viswa Vidyapeetam
kollam,kerala, India
amenu4aie20163@am.students.amrita.edu

V.Bharath Chandra Varma AM.EN.U4AIE20172
Department of Compter Science(Artificial Intelligence)
Amrita Viswa Vidyapeetam
kollam,kerala, India
amenu4aie20172@am.students.amrita.edu

***Abstract*—With the rapid development of Internet, image information is growing. It requires a lot of image storage and transmission. In order to reduce the storage and get better image quality, image compression algorithm is studied.It can beaccomplished by reducing the redundant and visually irrelevant information present in the images. There are many compression techniques to do image compression but in this paper we use a new image compression algorithm that uses principal component analysis (PCA), the image is decomposed by PCA.We use use for image compression because PCA has good image quality,compression ratio. Compressed images are evaluated based on the memory size reduction and percentage on how good it can summarize or explain the variance of the original image. The objective of image compression is to reduce the memory size to be as small as possible while maintaining the similarity with the original image. First the input image is compressed using PCA. Few of the Principal Components are used to compress the image. The reconstructed image obtained by inverse PCA.The reason behind constructiog reconstructed image is to know whether our above done pca process is corrrect or not.**

***Index Terms*—Compression techniques,Image quality,compression ratio,Variance,Principal component analysis(PCA)**

I. Introduction

A. Image Compression

Image compression is the process of encoding or converting an image file in such a way that it consumes less space than the original file. It is a type of compression technique that reduces the size of an image file without affecting or degrading its quality to a greateer extent.

Image compression may be lossy or lossless. Lossless compression is preferred for archival purposes and often for medical imaging, technical drawings, clip art, or comics. Lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossy methods are especially suitable for natural images such as photographs in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate. Lossy compression that produces negligible differences may be called visually lossless. And image compression is widely

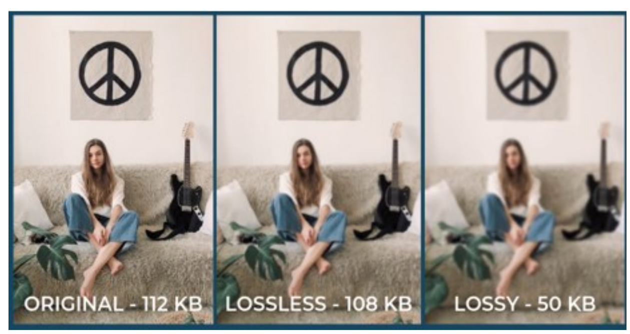


Fig. 1. Comparision

used in many applications we are explaining its one of the application below: The usage of medical images became necessary for the diagnosis of patients and hence large numbers of images are produced and used. Due to the large generation of medical images, it is very much essential to process the compression of images. Henceforth compression of medical images plays a vital role for storage and transmission.It is very important that while performing compression on the medical images, the effectiveness of resolution as well as perceptual quality be restored. There are different compression techniques uses different medical images like Magnetic resonance images (MRI) and X-ray angiograms (XA) etc. DICOM (digital imaging and communications in medicine) is used for storing, transmitting and viewing of the medical images.

And Nowadays there are many applications where the image compression tools used to effectively increased efficiency and performance. Applications like Health Industries, Retail Stores, Federal Government ,Agencies,Security Industries, Museums and Galleries

B. Principle Component Analysis

It is a dimensionality reduction technique that enables you to identify correlations and patterns in a data set so that it can be transformed into a data set of significantly lower dimension without loss of any important information. The below steps need to be followed to perform dimensionality reduction using PCA:

Standardization of the data
Computing the covariance matrix
Calculating the eigenvectors and eigenvalues
Choosing components and forming a feature vector:
Reducing the dimensions of the data set

Step 1: Standardization of the data The aim of this step is to standardize the data that we have so that PCA works properly. This is done by subtracting the respective means from the numbers in the respective column. So if we have two dimensions X and Y, all X become x-and all Y become y- This produces a dataset whose mean is zero.

Formula:

$$Z = \frac{Variable\ value\ -\ mean}{Standard\ deviation}$$

Step 2: Computing the covariance matrix As mentioned earlier, PCA helps to identify the correlation and dependencies among the features in a data set. A covariance matrix expresses the correlation between the different variables in the data set. It is essential to identify heavily dependent variables because they contain biased and redundant information which reduces the overall performance of the model. we are considering 2 dimensional input so we are taking X,Y axes Covariance formula:

$$Cov(x,y) = \frac{\Sigma\,(x_i - \bar{x}) * (y_i - \bar{y})}{(N-1)}$$

Covariance matrix format:

$$\begin{bmatrix} var(x) & cov(x,y) \\ cov(x,y) & var(y) \end{bmatrix}$$

Here, Var[X] = Cov[X,X] and Var[X] = Cov[X,X].

Step 3: Calculating the Eigenvectors and Eigenvalues An eigenvector is a nonzero vector that changes at most by a scalar factor when that linear transformation is applied to it. The corresponding eigenvalue is the factor by which the eigenvector is scaled. Let A be a square matrix (in our case the covariance matrix), ν a vector and λ a scalar that satisfies Aν = λ ν, then λ is called eigenvalue associated with eigenvector ν of A. formula for finding eigen values is:

$$\det(A - \lambda I) = 0$$

then its formula for finding eigen vectors is:

$$A\nu - \lambda\nu = 0\ ;\ (A - \lambda I)\nu = 0$$

Step 4: Choosing components and forming a feature vector: Once we have computed the Eigenvectors and eigenvalues, all we have to do is order them in the descending order, where the eigenvector with the highest eigenvalue is the most significant and thus forms the first principal component. The principal components of lesser significances can thus be removed in order to reduce the dimensions of the data. The final step in computing the Principal Components is to form a matrix known as the feature matrix that contains all the significant data variables that possess maximum information about the data.

The feature matrix for 2 dimension input is:

$$Feature\ Vector = (eig_1,\ eig_2)$$

Step 5: Reducing the dimensions of the data set: final step where we actually form the principal components using all the math we did till here. In order to replace the original data axis with the newly formed Principal Components, we simply multiply the transpose of the original data set by the transpose of the obtained feature vector.

$$NewData = FeatureVector^T \times ScaledData^T$$

Final data form:

Here, NewData is the Matrix consisting of the principal components, FeatureVector is the matrix we formed using the eigenvectors we chose to keep, and ScaledData is the scaled version of original dataset. This is the whole theory behind the entire PCA process.

II. Literature review

In 2012, stated that PCA used on wavelet coefficients to maximize edge energy in the reduced dimension images. Large image sets, for a better preservation of image local structures, a pixel and its nearest neighbors are modeled as a vector variable, whose training samples are selected from the local window by Local Pixel Grouping (LPG). In Muresan and Parks proposed a spatially adaptive principal component analysis (PCA) based denoising scheme. Elad and Aharon proposed sparse redundant representation and (clustering-singular value decomposition) K-SVD based denoising algorithm by training a highly over-complete dictionary. Foi et al. applied a shape-adaptive discrete cosine transform (DCT) to the neighborhood, which can achieve very sparse representation of the image and hence lead to effective denoising, recently Dabov et al. proposed a collaborative image denoising scheme by patch matching and sparse 3D transform. They searched for similar blocks in the image by using block matching and grouped those blocks into a 3D cube.

III. Dataset

A data set is a collection of data.We have taken the dataset from a open source website known as benchmark by Rawzor.Name of our dataset is RGB16BIT.In our dataset we have 14 color images. The data set which we are using in this experiment to do image compression is deer.jpg. Intially the size of big building image is 5.88mb,by using image compression using pca we are reducing the size of the image,Basically when we can reduce upto very small memory storage by selecting less principal components but according to studies if we try to compress the image more than 40percent of its original pixels(or)principal components then the information of the image will not be appropriate. so we are taking 1700 principal components of the big building image. the reason behind taking 1700 principal components is intially,our image is of 4046 principal components as i mentioned above we should extract images below 40percent of their pixels(or)principal components so,40percent of 4046 is 1618 so,we are taking that number of principal components. so that the information of image wont be poor,it can be informative. And if we observe the the size of compressed image is 4.27mb its less than the source image so we got our required output.

IV. Methods

Step1: Taking the Data
Step2:Computing co-variance matrix
Step3:Calculating the Eigen vectors
Step4:Choosing components and forming a feature vector
Step5:Reducing the dimensions of the data set
Step6:Evaluating the Processed image by PSNR and MSE.

A. Taking the data

Step1:Take the image as input(the input is an rgb image)
Step2:Convert that rgb image to to gray scale;
Step3:Change the datatype to double

B. Computing co-variance matrix

Covarieance matrix is a symmetric matrix that shows covariances of each pair of variables. These values in the covariance matrix show the distribution magnitude and direction of multivariate data in multidimensional space. By controlling these values we can have information about how data spread among two dimensions.

Step1:Compute the mean of the value of input image
Step2:Subtract the mean from each datapoint
Step3:use cov function in matlab to fing the covariance matrix

C. Calculating Eigen vectors

Eigenvectors and eigenvalues have many important applications in computer vision and machine learning in general. Well known examples are PCA (Principal Component Analysis) for dimensionality reduction or EigenFaces for face recognition.An eigenvector is a vector whose direction remains unchanged when a linear transformation is applied to it.

Step1:Use that covarience matrix and compute the eigan vector by the processes described in Introduction part.But in code we can do it by inbuilt function called eig.

Step2:From that eig function we will return the eigen vectors.

D. Calculating the Principle components

The principle components are calculated by multiplying eigen vectors which is returned and the difference between the data and mean.Through this we will get the principle component values

E. Reducing the principle components

In this we will take the no of Principle Components as input and then through for loop we will change the corresponding eigen vectors and we'll do the above process again.

F. Evaluation of PSNR and MSE

CALCULATION OF MSE

$MSE = sum(sum((Dataofgrayscaleinputimage - Dataofcompressedimage).^2))/(M*N)$ Through the above formula we can find the mean Squared error

CALCULATION OF PSNR

PSNR = 10*log10(256*256/MSE);

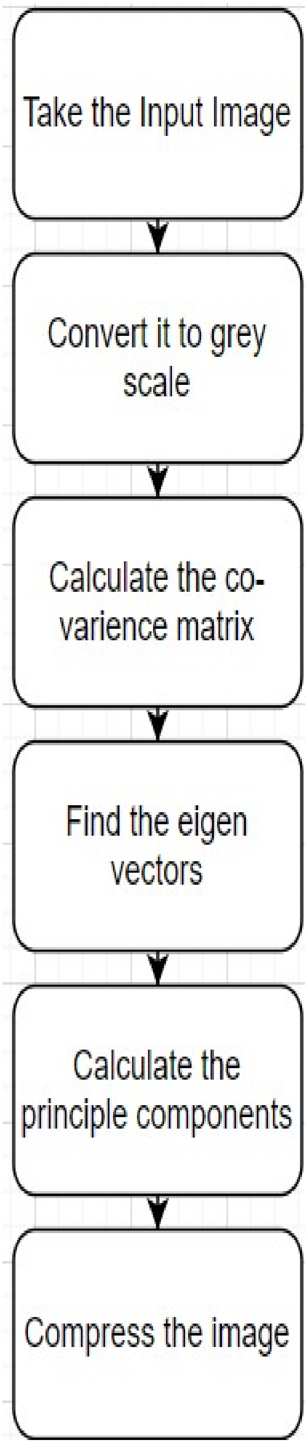
we will subtitute the calculates MSE in above formula to get the PSNR value.

CALCULATION OF COMPRESSION RATIO(CR)

CR will be the ration between the size of compressed image to the size of Input image.(the sizes are taken in bites)

Through the above three evaluations we can conclude how much the image is compressed.

G. *Flowchart for Method of solving*

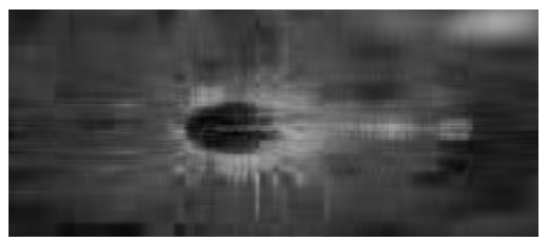


V. RESULTS

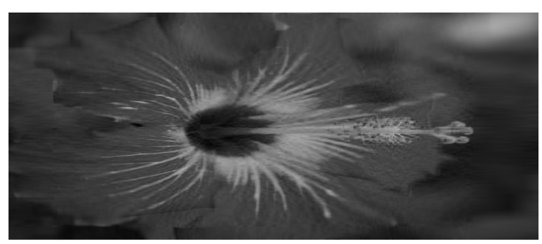
A. *RBG image*

B. *Grey Scale image*

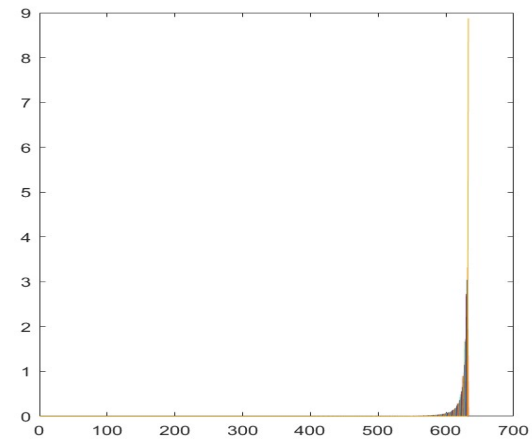
C. *For 10 PCs*



D. *For 50 PCs*



E. *For 100 PCs*



F. plot for Eigen values

G. Conclusion

We have done Image compression using PCA

We took flower image as input.

We could enter the required principle components and get the required compressed image.

So, from this project we've got the clarity about PCA and image compression.

References

[1] Arab, Abbas, et al. "Image Compression Using Principle Component Analysis." Al-Mustansiriyah Journal of Science, vol. 29, no. 2, 2018, p. 141., doi:10.23851/mjs.v29i2.256.

[2] Santo, Rafael Do Espírito. "Principal Component Analysis Applied to Digital Image Compression." Einstein (São Paulo), vol. 10, no. 2, 2012, pp. 135–139., doi:10.1590/s1679-45082012000200004.

[3] Pramoditha, Rukshan. "Image Compression Using Principal Component Analysis (PCA)." Medium, Towards Data Science, 13 Apr. 2021, towardsdatascience.com/image-compression-using-principal-component-analysis-pca-253f26740a9f.

[4] "PCA in Machine Learning: Assumptions, Steps to Apply amp; Applications." UpGrad Blog, 12 Nov. 2020, www.upgrad.com/blog/pca-in-machine-learning/.

[5] "Psnr." Compute Peak Signal-to-Noise Ratio (PSNR) between Images - Simulink - MathWorks India, in.mathworks.com/help/vision/ref/psnr.html.

[6] "Lossy Plus Lossless Residual Encoding." Digital Image Compression Techniques, pp. 68–69., doi:10.1117/3.34917.ch8.