Principal Component Analysis using Singular Value Decomposition for Image Compression

Prasannajit Dash
Assistant Professor
Orissa Engineering College,
BPUT, Bhubaneswar, India

Maya Nayak, Ph.D
Professor
Orissa Engineering College,
BPUT,Bhubaneswar,India

Guru Prasad Das
Lecturer
Orissa Engineering College,
BPUT,Bhubaneswar, India

ABSTRACT

Principal components analysis (PCA) is one of a family of techniques for taking high-dimensional data, and using the dependencies between the variables to represent it in a more tractable, lower-dimensional form, without losing too much information. PCA is one of the simplest and most robust ways of doing such dimensionality reduction. It is also one of the best, and has been rediscovered many times in many fields, so it is also known as the Karhunen-Lo_eve transformation, the Hotelling transformation, the method of empirical orthogonal functions, and singular value decomposition.

General Terms

Variances, covariance, symmetric matrix, identity matrix, orthogonal matrix, diagonal matrix

Keywords

Principal Component Analysis (PCA), Singular Value decomposition (SVD)

1. INTRODUCTION

Assume the data set is represented in terms of m×n matrix. Let the data set is X where m is considered as columns of the samples i.e. observations and n is considered as the variables. To transform the matrix in linear form i.e. X to another matrix Y having same dimension i.e. m×n, so that for some m×m matrix P i.e.

$$Y = P \times X \qquad (1)$$

If rows of P will be considered as row vector i.e. $p_1, p_2, p_3, \ldots, p_m$ and columns of X will be considered as column vectors i.e. $x_1, x_2, x_3, \ldots, x_n$, then PX can be interpreted as below :

$$Y = P \times X = \begin{pmatrix} p_1 \times x_1 \, p_1 \times x_2 \ldots \ldots p_1 \times x_n \\ p_2 \times x_1 \, p_2 \times x_2 \ldots \ldots p_2 \times x_n \\ \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \\ p_m \times x_1 \, p_m \times x_2 \ldots \ldots p_m \times x_n \end{pmatrix}$$

$$(2)$$

As $p_i, x_j \in R^{m \times n}$ and so $p_i . x_j$ is the Euclidean inner (dot) product. This gives the original data set X, being projected to the columns of P i.e. $(p_1, p_2, \ldots, p_m)$ which formulates the new basis for representing the columns of X. The rows of P will later become the Principal Components (PCs). Principal Component Analysis [1-3] defines independence by calculating the variances of data in original basis. It seeks to de-correlate the original data by finding the directions in which variance is maximized and then use these directions to define the new basis. The definition for the variance of a random variable, Z with mean, μ is as below:

$$\sigma^2 Z = E\left[(Z - \mu)^2 \right] \qquad (3)$$

Suppose consider a vector of n discrete measurements, i.e. r̃ = (r̃_1, r̃_2 . . . r̃n), with mean μ_r. If the mean is subtracted from each of the measurements, then a translated set of measurements r = (r_1, r_2, . . . , rn), that has zero mean. Thus, the variance of these measurements is given by the relation

$$\sigma^2 r = \frac{1}{n} \times \left(r \times r^T \right) \qquad (4)$$

If a second vector of n measurements, i.e. s = (s_1, s_2, . . . , s_n), again with zero mean, then idea comes to obtain the covariance of r and s. Covariance can be thought of as a measure of how much two variables change together. Variance is thus a special case of covariance, when the two variables are identical. It is in fact correct to divide through by a factor of (n – 1) rather than n. The original m×n data matrix i.e. X having m number of variables and n number of samples. Hence X can be considered m row vectors of each of length n.

$$X = \begin{pmatrix} x_{1,1} x_{1,2} x_{1,3} \ldots \ldots x_{1,n} \\ x_{2,1} x_{2,2} x_{2,3} \ldots x_{2,n} \\ \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \\ x_{m,1} x_{m,2} x_{m,3} \ldots \ldots x_{m,n} \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} x_1 \\ x_2 \\ \ldots \\ x_m \end{pmatrix} \in R^{m,n}, x_i^T \in R^n$$

$$(5)$$

Since there is a row vector of each variable where each of these vector consists of all samples of a particular variable. For example, x_i is a vector of the 'n' samples for the i_{th} variable.

$$C_x = \frac{1}{(n-1)} \times \left(XX^{\mathrm{T}}\right) \in R^{m \times m}$$

$$\Rightarrow \frac{1}{(n-1)} \times \begin{pmatrix} x_{1 \times 1}{}^{\mathrm{T}} x_{1 \times 2}{}^{\mathrm{T}} \ldots x_{1 \times m}{}^{\mathrm{T}} \\ x_{2 \times 1}{}^{\mathrm{T}} x_{2 \times 2}{}^{\mathrm{T}} \ldots x_{2 \times m}{}^{\mathrm{T}} \\ \ldots\ldots\ldots\ldots\ldots\ldots \\ x_{m \times 1}{}^{\mathrm{T}} x_{m \times 2}{}^{\mathrm{T}} \ldots x_{m \times m}{}^{\mathrm{T}} . \end{pmatrix}$$

(6)

This matrix is therefore known as the Covariance Matrix. Covariance is a measure of how well correlated two variables are. The PCA [4-6] method makes the fundamental assumption that the variables in the transformed x should be as uncorrelated as possible. It means covariances of different variables in the matrix i.e. C_Y, should be as close to zero as possible. It gives the idea i.e. covariance matrices are always positive definite or positive semi-definite. Therefore the following requirements for constructing the covariance matrix, Cy:

(a) In order to maximize the diagonal entries gives the measurement as variance.

(b) In order to minimize the off-diagonal entries gives the covariance between the variables.

Consider the formula for constructing the covariance matrix i.e. Cy where the previous interpretation of matrix Y against the original data set matrix X given in Eq. (1) as below:

$$C_y = \frac{1}{(n-1)} \times \left(YY^{\mathrm{T}}\right) \Rightarrow \frac{1}{(n-1)} \times \left((PX)(PX)^{\mathrm{T}}\right)$$

$$\Rightarrow \frac{1}{(n-1)} \times \left((PX)(X^{\mathrm{T}}P^{\mathrm{T}})\right) \Rightarrow \frac{1}{(n-1)} \times \left(PXX^{\mathrm{T}}P^{\mathrm{T}}\right)$$

$$\Rightarrow \frac{1}{(n-1)} \times \left(PSP^{\mathrm{T}}\right) where S = \left(XX^{\mathrm{T}}\right)$$

$$\Rightarrow C_y = \frac{1}{(n-1)} \times \left(PSP^{\mathrm{T}}\right)$$

(7)

As the matrix S is in order of m × m, the matrix can be square symmetric if

$$\left(XX^{\mathrm{T}}\right)^{\mathrm{T}} = \left((X^{\mathrm{T}})^{\mathrm{T}} \times X^{\mathrm{T}}\right) \Rightarrow \left(XX^{\mathrm{T}}\right)$$ (8)

From linear algebraic theorem, it is well known that for a square symmetric matrix [11] which is orthogonally diagnosable. Hence the following matrix S can be written as below:

$$S = EDE^{\mathrm{T}}$$ (9)

where E is the orthogonal matrix of m × m order whose columns are the eigenvectors of the matrix S and D is the diagonal matrix of having eigenvalues of matrix S. Let the transformation matrix P consists of rows as eigenvectors of the matrix S and columns as eigenvalues of matrix S. So P = E^{T}. Hence the covariance matrix Cy is reconsidered as

$$C_y = \frac{1}{(n-1)} \times \left(PSP^{\mathrm{T}}\right) \Rightarrow \frac{1}{(n-1)} \times \left(E^{\mathrm{T}}\left(EDE^{\mathrm{T}}\right)E\right)$$

$$\Rightarrow \frac{1}{(n-1)} \times \left((E^{\mathrm{T}}E)(D)(E^{\mathrm{T}}E)\right) \Rightarrow \frac{1}{(n-1)} \times (I \times D \times I)$$

(10)

where I is the identity matrix of order m×m.

Hence for the special choice of P, we gain

$$C_y = \frac{1}{(n-1)} \times D$$ (11)

Hence it is concluded that the largest variance goes to the first principal components, the second largest variance goes to the second principal components and so on. Therefore it gives us a method for organizing the data in the diagonalisation stage. Once it produces the eigenvalues and eigenvectors of S i.e. XX^{T}, it is sorted with the eigenvalues in descending order and place them in this order on the diagonal of D. So the orthogonal matrix[13], E is created by placing the associated eigenvectors in the same order to form the columns of E (i.e. place the eigenvector that corresponds to the largest eigenvalue in the first column, the eigenvector corresponding to the second largest eigenvalue in the second column etc.).Therefore the objective of diagonalising the covariance matrix of the transformed data is achieved. The principal components (the rows of P) are the eigenvectors of the covariance matrix, XX^{T}, and the rows are in order of 'importance', telling how 'principal' each principal component is.

2. SINGULAR VALUE DECOMPOSITION TECHNIQUE

This section adds the benefit of Singular Value Decomposition algorithm[8] in linear algebra to the Principal Component Analysis[7]. Let $R^{n \times n}$, $R^{n \times m}$ and $R^{m \times m}$ are three vector spaces where $R^{n \times n}$ is orthogonal, $R^{n \times m}$ is diagonal and $R^{m \times m}$ is again the orthogonal one. Let U be the orthogonal matrix belonging to the vector space $R^{n \times n}$, $\sum$ be the diagonal matrix to the vector space $R^{n \times m}$ and the V be orthogonal matrix to the vector space $R^{m \times m}$.

Given the matrix A i.e. A ϵ $R^{n \times m}$ (diagonal), the singular value decomposition of A is as below:

$$A = U \times \sum \times V^{\mathrm{T}}$$ (12)

The diagonal entries i.e. σ_i ϵ $\sum$ is positive, then only the entries of the matrix A will be called the Singular Value Decomposition (SVD) [8]. The diagonal entries are ordered so that the largest diagonal entry i.e. σ_1 is placed in (1, 1) position of the matrix A. Subsequently the second largest entries i.e. σ_2 is placed in (2, 2) and so on till the end diagonal point. The reversed row and column indexes in defining the SVD from the way they were defined in the derivation of PCA [7] makes it become apparent.

Since U ϵ $R^{n \times n}$ and V ϵ $R^{m \times m}$ are orthogonal matrices, their columns form bases for the vector spaces R^n and R^m. Therefore, any vector b ϵ R^n can be expanded in the basis formed by the columns of U (also known as the left singular vectors of A) and any vector x ϵ R^m can be expanded in the basis formed by the columns of V (also known as the right singular vectors of A). The vectors for these expansions ˆb and ˆx, are given by:

$$\tilde{b} = U^{\mathrm{T}} \times b \Rightarrow \tilde{x} = V^{\mathrm{T}} \times X$$ (13)

If the relation b = (A × x) then the conclusion is as below:

$$\tilde{b} = \left(U^{\mathrm{T}} \times b\right) \Rightarrow \left(U^{\mathrm{T}} \times (A \times x)\right)$$
$$\Rightarrow \left(U^{\mathrm{T}} \times \left(U \times \Sigma \times V^{\mathrm{T}}\right) \times x\right)$$
$$\Rightarrow \left(U^{\mathrm{T}} \times U\right) \times \left(\Sigma \times V^{\mathrm{T}} \times x\right) \Rightarrow 1 \times \left(\Sigma \times \tilde{x}\right)$$
$$\Rightarrow \left(\Sigma \times \tilde{x}\right)$$

(14)

Thus, the SVD allows us to assert that every matrix is diagonal following the above rules. In a link to Principal Component Analysis, there is a theorem in linear algebra which says that every non-zero singular values of matrix A $(U \times \Sigma \times V^{\mathrm{T}})$ is the square roots of the eigenvalues of $A^{\mathrm{T}} \times A$ so that former assertion becomes true.

$$A^{\mathrm{T}} \times A = \left(U \times \Sigma \times V^{\mathrm{T}}\right)^{\mathrm{T}} \times \left(U \times \Sigma \times V^{\mathrm{T}}\right)$$
$$\Rightarrow \left(V \times \Sigma^{\mathrm{T}} \times U^{\mathrm{T}}\right) \times \left(U \times \Sigma \times V^{\mathrm{T}}\right)$$
$$\Rightarrow (V \times \left(\Sigma^{\mathrm{T}} \times \Sigma\right) \times V^{\mathrm{T}})$$

(15)

Since the covariance matrix $A^{\mathrm{T}} \times A$ is a square symmetric, it presents the orthogonal diagonalization where the eigenvectors are columns of orthogonal matrix V. This is very important in making the practical connection between Singular value Decomposition (SVD) and Principal Component Analysis (PCA) of the original matrix A. Since there is the original m × n data matrix, i.e. X, let's define a new (n × m) matrix, Z:

i.e. $z = \dfrac{1}{\left(\sqrt{(n-1)}\right)} \times X^{\mathrm{T}}$ (16)

As the m rows of X contained the n data samples, let subtract the row average from each row entry to ensure zero mean across the rows. Thus, the new matrix, Z has the columns with zero mean. Then forming the m × m matrix,

$$Z^{\mathrm{T}} \times Z = \left(\frac{1}{\left(\sqrt{n-1}\right)} \times X^{\mathrm{T}}\right)^{\mathrm{T}} \times \left(\frac{1}{\left(\sqrt{(n-1)}\right)} \times X^{\mathrm{T}}\right)$$
$$\Rightarrow \left(\frac{1}{\left(\sqrt{(n-1)}\right)} \times X\right) \times \left(\frac{1}{\left(\sqrt{(n-1)}\right)} \times X^{\mathrm{T}}\right)$$
$$\Rightarrow \frac{1}{(n-1)} \times XX^{\mathrm{T}}$$

(17)

i.e. $Z^{\mathrm{T}} \times Z = C_{x}$ (18)

as $C_{x} = \left(\dfrac{1}{(n-1)}\right) \times XX^{\mathrm{T}}$ (19)

It has been found that by defining Z in this way ensures that $Z^{\mathrm{T}} \times Z$ is equal to the covariance matrix of X, i.e., C_{X}. So it has been identified that the principal components of X are the eigenvectors of C_{X}. Therefore, by performing a singular value decomposition of the matrix $Z^{\mathrm{T}} \times Z$, the principal components will be the columns of the orthogonal matrix, V.

3. IMAGE COMPRESSION USING PCA

As it has been developed that the method of singular value decomposition(SVD) in principal component analysis i.e. $Z^{\mathrm{T}} \times Z$ which is equal to the covariance matrix Cx derived from the original data set matrix of m×n order i.e. X. By recalling the orthogonal matrix V where the relation says that Y = $V^{\mathrm{T}} \times$ X, but we need to truncate the orthogonal matrix V where we need to select the first 'r' rows out of 'm' rows so that one new matrix V1 can be considered as :

V1 ϵ $R^{m \times r}$ (orthogonal vector space)

Though the projection Y1 = V1 × X where V1 is of the order i.e. m × r and the original matrix X of order m × n, so that the projected matrix Y1 will have r × n order i.e. Y1 ϵ $R^{r \times n}$. Hence the newly projected matrix i.e. Y1 is still dimensionally consistent. Suppose in order to transform the data to the original basis of the original data set X by computing i.e.

$$X = Y1 \times V1$$ (20)

Hence the dimensions of the original data matrix X are recovered, i.e. we obtain X1 ϵ $R^{m \times n}$. The matrices X and X1 are of the same dimensions, but they are not the same matrix, since it has been truncated the matrix of principal dimensions of the color image is 3216 × 2136 where width is 3216 pixels, height is 2136 pixels, and horizontal resolution is 300 dpi, vertical components V in order to obtain X1. It is therefore reasonable to conclude that the matrix, X1 has in some sense, 'less information' in it than the matrix X. Load the color image in Figure 1. into MATLAB. The resolution is 300 dpi, bit depth is 24 and the size is 3.32MB.

The RGB source color image(JPEG format) is stored in matrix of 2136 × 3216 × 3 order, where the third dimension stores three numbers in the range[0,1] corresponding to each pixel. The third dimension stores the numbers representing the intensity of Red, Green and Blue (RGB). The RGB image is further converted to gray-scale image as in Figure 2 which produces the matrix of exact 2136 × 3216 × 3 order as the size got reduced to 93 KB as usual but the gray scale image is the image need to go compression through Principal Component Analysis approach where the third dimension represents the intensity need to undergo dimensionality reduction to certain compression ratio. The colourmap matrix of order 2136 × 3216 × 3 with three identical columns with each of a scale representing intensity on the one dimensional gray scale. It means each element of the pixel or dot matrix contains a number representing a certain intensity of an individual pixel or dot of gray scale image. Let the data matrix i.e. X of order 2136 × 3216 which will undergo a principal component analysis(PCA)[7] using Singular Value Decomposition(SVD) method[8]. The image size is calculated as row value of 2136 pixels and column value of 3126 pixels. As the image is two-dimensional, the row mean is computed giving one dimensional 2136 × 1 matrix. Row mean is subtracted from each row element giving the changed matrix X having the same order of 2136 × 3216. Further a temporary matrix Z is created by multiplying

$1/(\sqrt{(n-1)} \times$ X' where n = 3216. So the order of transformed matrix Z is 3216 × 2136.

As the transformed or resulting matrix is two-dimensional where covariance matrix of Z is calculated as Z'× Z resulting matrix of order 2136 × 2136 pixels. Let say the covariance matrix as Cx. Computing the Singular Value Decomposition (SVD) [8] passing the covariance matrix i.e. Cx resulting three matrices as U, S and V. U is an orthogonal matrix of order 2136 × 2136 pixels, S is a diagonal matrix of order 2136 × 2136 where the matrix S contains all zeroes except the diagonal entries. The third matrix V is same as of U both element wise and order wise. Hence the variances is computed

by taking only the diagonal values of S resulting one dimensional matrix of order 2136 × 1. So the cumulative variances got calculated in accordance with the above variances against the total sum of variances resulting 1 to 2136 values. Consider the first twenty Eigen values out of 2136 and calculate the Eigen vectors resulting the Eigen spectrum as in Figure 4. Let consider to use first 40 Principal Components (PCs) out of 2136 PCs. In order to know the compression ratio of first 40 PCs, we need to compare two matrices i.e. the original matrix X of order 2136 × 3126 without compression against the covariance matrix Cx of order 2136 × 2136. Since we select the first 40 Principal Components, the resulting matrices are two i.e. the first one of 2136 × 40 order and the other one is of 40 × 2136 order. There is also one more matrix which is must used in order to display the recovered image i.e. the vector of means in Figure 3. Therefore the image is reduced with the number of columns needed from 2136 to 40 + 40 + 1 = 41 and the compression ratio is then calculated in the following way:

2136/2(40 + 1) = 26.3704 where the 40 is accountable for first 40 Principal Components (PCs). So 2136: 81 = 26.3704: 1, a decent ratio it seems. The image for 40 principal components (26.3704: 1) is displayed in Figure 3. In Figure 5, a selection of images in increasing number of principal components corresponding to the compression ratio is displayed.

4. CONCLUSION

The loss in quality is evident, after all this lossy compression, as opposed to lossless compression considering the compression ratio seems quite very good. It has been observed that PCA's Singular Value Decomposition method makes tremendous drop in compression ratio against the increasing number of principal components for example. as in Table1 and Figure6. PCA is useful for finding new, more informative, uncorrelated features as it reduces dimensionality by rejecting low variances features. But in Independent Component Analysis, an independence condition is optimized which gives more meaning full components than optimization of only the variance as is done by PCA.

5. REFERENCES

[1] Shyi-Chyi Cheng, Shih-Chang Hsia, Fast algorithms for color image processing by principal component analysis Visual Communication and Image Representation, Volume 14, Issue 2, June 2003, Pages 184-203

[2] Ze Wang, Chi-Sing Leung, Tien-Tsin Wong, Yi-Sheng Zhu, Eigen-image based compression for the image-based relighting with cascade recursive least squared networks

Pattern Recognition, Volume 37, Issue 6, June 2004, Pages 1219-1231

[3] Clifford Clausen, Harry Wechsler, Color image compression using PCA and backpropagation learning Pattern Recognition, Volume 33, Issue 9, September 2000, Pages 1555-1560

[4] Sheng Liu, Mingming Gu, Qingchun Zhang, Bing Li, Principal component analysis algorithm in video compressed sensing Optik - International Journal for Light and Electron Optics, Volume 125, Issue 3, February 2014, Pages 1149-1153

[5] Daoqiang Zhang, Songcan Chen, Fast image compression using matrix K–L transform

Neurocomputing, Volume 68, October 2005, Pages 258-266

[6] A. Zabala, X. Pons,Effects of lossy compression on remote sensing image classification of forest areas International Journal of Applied Earth Observation and Geoinformation, Volume 13, Issue 1, February 2011, Pages 43-51

[7] Kuo-Feng Hwang, Chin-Chen Chang, A fast pixel mapping algorithm using principal component analysis Pattern Recognition Letters, Volume 23, Issue 14, December 2002, Pages 1747-1753

[8] Jan J. Gerbrands, On the relationships between SVD, KLT and PCA Pattern Recognition, Volume 14, Issues 1–6, 1981, Pages 375-381

[9] J. Jiang, Image compression with neural networks – A survey Signal Processing: Image Communication, Volume 14, Issue 9, July 1999, Pages 737-760

[10] Zachary B. Charles, Miriam Farber, Charles R. Johnson, Lee Kennedy-Shaffer, The relation between the diagonal entries and the eigenvalues of a symmetric matrix, based upon the sign pattern of its off-diagonal entries Linear Algebra and its Applications, Volume 438, Issue 3, 1 February 2013, Pages 1427-1445

Figure1. Source Color Image of a temple
in RGB color format

Figure2. Gray scale image of the temple
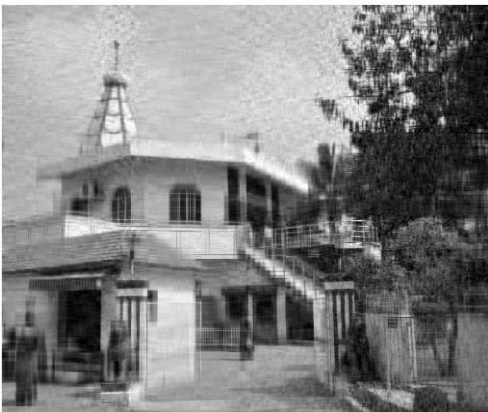converted from the RGB color format

Figure3. Recovered image of the temple
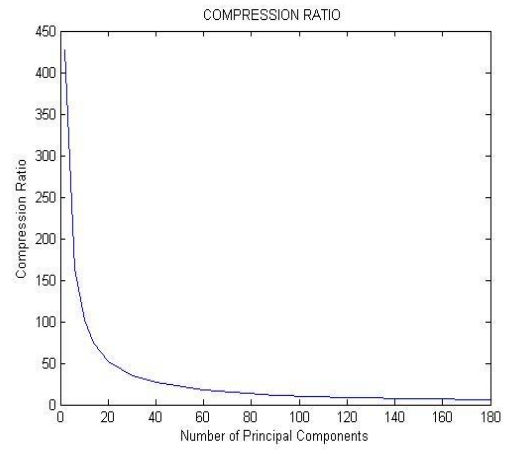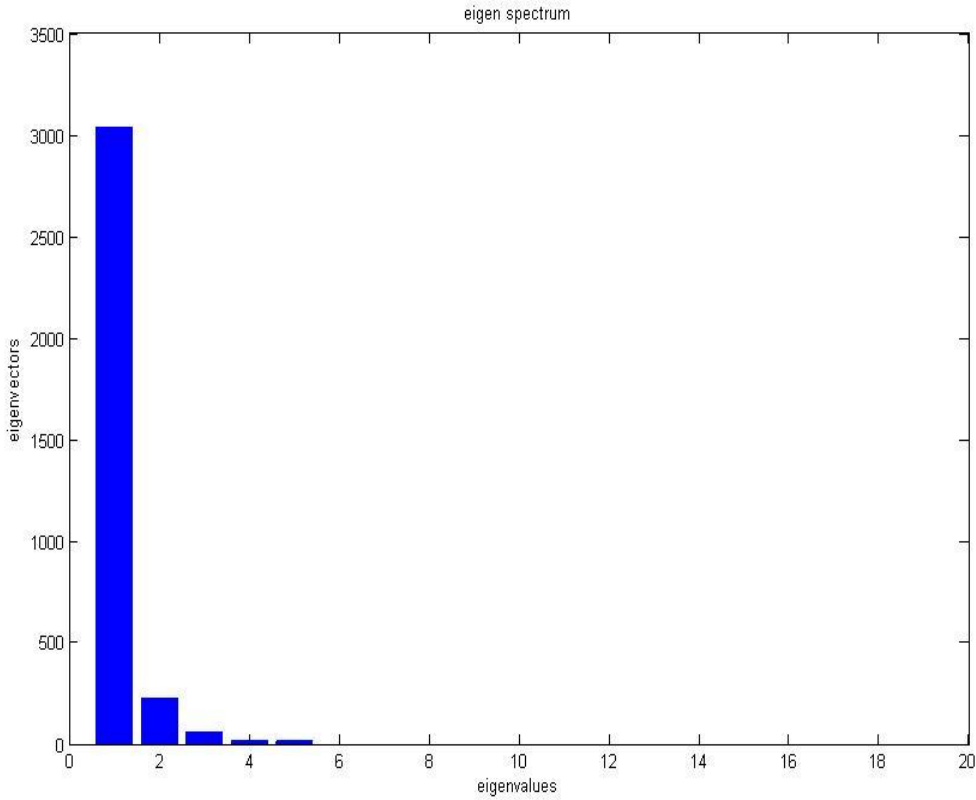after data and dimensionality reduction

Figure6. Comparing compression ratio against 0 to
180 Principal Components

Figure4. The first twenty Eigen Values out of 2136 and the Eigen Vectors resulting the Eigen Spectrum

Table 1 Compression Ratio in respect to the number of Principal Components

SL NO.	NO. OF PRINCIPAL COMPONENTS	COMPRESSION RATIO
1	2 Principal Components	427.2:1 compression
2	6 Principal Components	163.3:1 compression
3	10 Principal Components	101.7:1 compression
4	14 Principal Components	73.7:1 compression
5	20 Principal Components	52.1:1 compression
6	30 Principal Components	35:1 compression
7	40 Principal Components	26.4:1 compression
8	60 Principal Components	17.7:1 compression
9	90 Principal Components	11.8:1 compression
10	120 Principal Components	8.9:1 compression
11	150 Principal Components	7.1:1 compression
12	180 Principal Components	5.9:1 compression

427.2:1 compression
2 principal components

164.3:1 compression
6 principal components

101.7:1 compression
10 principal components

73.7:1 compression
14 principal components

52.1:1 compression
20 principal components

35:1 compression
30 principal components

26.4:1 compression
40 principal components

17.7:1 compression
60 principal components

11.8:1 compression
90 principal components

8.9:1 compression
120 principal components

7.1:1 compression
150 principal components

5.9:1 compression
180 principal components

Figure5. A selection of images in increasing number of principal components corresponding to the compression ratio is displayed.