

Classification of Image Database Using Independent Principal Component Analysis

H. B. Kekre

Professor,
Computer Engineering
Mukesh Patel School of Technology,
Management and Engineering,
NMIMS University, Vileparle(w)
Mumbai 400-056, India

Tanuja K. Sarode

Associate Professor,
Computer Engineering,
Thadomal Shahani Engineering
College, Bandra(W),
Mumbai 400-050, India

Jagruti K. Save

Ph.D. Scholar, MPSTME,
NMIMS University,
Associate Professor,
Fr. C. Rodrigues College of
Engineering, Bandra(W),
Mumbai 400-050, India

Abstract—the paper presents a modified approach of Principal Component Analysis (PCA) for an automatic classification of image database. Principal components are the distinctive or peculiar features of an image. PCA also holds information regarding the structure of data. PCA can be applied to all training images of different classes together forming universal subspace or to an individual image forming an object subspace. But if PCA is applied independently on the different classes of objects, the main direction will be different for them. Thus, they can be used to construct a classifier which uses them to make decisions regarding the class. Also the dimension reduction of feature vector is possible. Initially training image set is chosen for each class. PCA, using eigen vector decomposition, is applied to an individual class forming an individual and independent eigenspace for that class. If there are n classes of training images, we get n eigenspaces. The dimension of eigenspace depends upon the number of selected eigen vectors. Each training image is projected on the corresponding eigenspace giving its feature vector. Thus n sets of training feature vectors are produced. In testing phase, new image is projected on all eigenspaces forming n feature vectors. These feature vectors are compared with training feature vectors in corresponding eigenspace. Feature vector nearest to new image in each eigenspace is found out. Classification of new image is accomplished by comparing the distances between the nearest feature vectors and training image feature vector in each eigenspace. Two distance criteria such as Euclidean and Manhattan distance are used. The system is tested on COIL-100 database. Performance is tested and tabulated for different sizes of training image database.

Keywords—Image classification; Principal Component Analysis (PCA); Eigen value; Eigen vector; Variance; Nearest neighbor classifier; Orthogonal transform; Feature vector; Covariance matrix.

I. INTRODUCTION

Principal Component Analysis (PCA) is the most popular and the oldest multivariate statistical technique [1]. PCA was invented in 1901 by Karl Pearson [2], who formulated the analysis as "Finding lines and planes of closest fit to systems of points in space". The focus was on geometric optimization. Later it was re-invented by Harold Hotelling in 1933[3]. In image analysis, the term Hotelling transformation is often used for a principal component projection. PCA is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since

patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analysing data.

Generally classification of images is a two step process, feature vector generation followed by a nearest-neighbor classifier [4][5]. Classification accuracy depends on many factors. One major factor is the extraction of features to represent the image. Feature extraction is a special form of dimensionality reduction. PCA is a method used to reduce the number of features used to represent the data. The benefits of this dimensionality reduction include providing a simpler representation of the data, reduction in memory, and faster classification. PCA transforms the original variables to a new set of variables, that are uncorrelated and ordered such that the first few retains most of the information present in the data.[6]. These uncorrelated components are called principal components (PC) and are estimated from the eigenvectors of the covariance or correlation matrix of the original variables. PCA has been widely used for image processing applications such as face recognition[7][8][9][10], palm print recognition[11], image compression[12][13], image fusion[14], image enhancement[15] [16], object recognition[17] etc.

This research paper is structured as follows: Section II explains the generation of principal components by PCA method. Section III describes the methodologies of the system. Section IV presents the results. Finally section V describes the conclusions and proposing some possible future work followed by references.

II. PRINCIPAL COMPONENT ANALYSIS

Principle Components Analysis (PCA) is a well-known method to identify statistical trends in data. It projects the data from a higher dimension to a lower dimensional manifold such that the error incurred by reconstructing the data in the higher dimension is minimized. As shown in fig. 1, given a set of points in Euclidean space, the first principal component Z_1 corresponds to a line that passes through the multidimensional mean and minimizes the sum of squares of the distances of the points from the line. The second principal component Z_2 corresponds to the same concept after all correlation with the first principal component has been subtracted from the points. Principal components are a series of linear least square fits to a sample, each orthogonal to all previous. The Principle

Components reveal important information about the dispersion of the original data set.

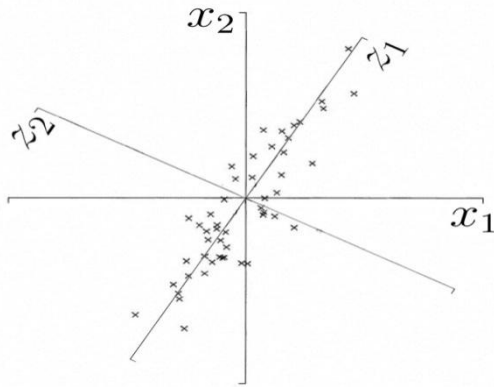


Fig.1. Principal components Z_1 and Z_2

PCA is based upon eigenvector decomposition of a covariance matrix. For multivariate data, covariance is a measure of the relationship between different variables, or dimensions of the data set. The general steps of PCA[18][19] are as follows:

- 1) Acquire data.
- 2) Subtract mean from the data.
- 3) Generate the covariance matrix[20]. Important property of covariance matrix is that it is square, real, and symmetric. This means that there always exists n real eigenvalues for an $n \times n$ covariance matrix.
- 4) Calculate the eigenvalues and eigenvectors of the covariance matrix. The first Principle Component is the eigenvector of the covariance matrix with the largest eigenvalue. It represents the most significant relationship between the data dimensions.
- 5) Compute the cumulative energy content for each eigenvector.
- 6) By ordering the eigenvectors in the order of descending eigenvalues (largest first), create an ordered orthogonal basis.
- 7) Use this basis to transform input data vector. Instead of using all the eigenvectors of the covariance matrix, the data can be represented in terms of only a few basis vectors of the orthogonal basis.

III. PROPOSED ALGORITHM

From the image database, some images are used for training and the remaining images are used for testing. The algorithm used to generate the feature vector for each training image is given below.

A. Feature vector generation for training images

Consider there are 'M' training images in each class and there are such 'N' classes. All images are converted into gray scale images.

For each class do the steps 1 to 10.

Step 1: Find the average image of that class. Refer equation 1.

$$I_{avg}(x, y) = \frac{1}{M} \sum_{i=1}^M I_i(x, y) \quad (1)$$

Step 2: Find zero mean images by subtracting the average image from each image as given in equation 2.

$$I_i = I_i - I_{avg} \quad (2)$$

for $i = 1 \dots M$

Step 3: Convert zero mean images into one dimensional vector by arranging the columns of an image one below the other as shown in fig.2.

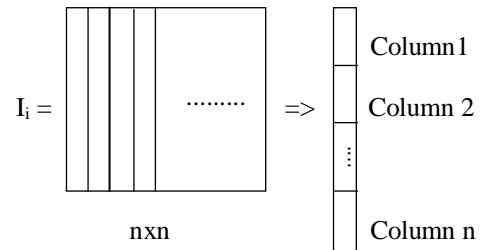


Fig.2. Conversion of an image into vector

Step 4: Organize these vectors say ϕ_i (for $i=1$ to M) in matrix form as shown below in fig.3.

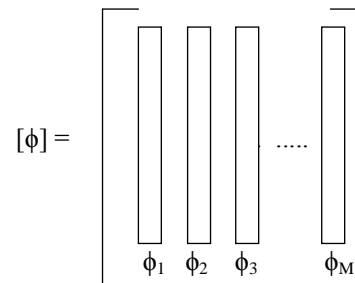


Fig.3. Organization of vectors

Step 5: Calculate the covariance matrix $A(M \times M)$ using the formula (refer equation 3).

$$A = 1/M \{ [\phi]^t [\phi] \} \quad (3)$$

Step 6: Calculate the eigen values (λ_1 to λ_M) and eigen vectors (X_1 to X_M) by solving equation 4. Eigen vectors are ordered according to the corresponding eigen values from high to low.

$$[A - \lambda I]X = 0 \quad (4)$$

Step 7: Construct eigen images as given in equation 5. (Exclude eigen vector corresponding to lowest eigen value as it is extremely small comparatively)

$$[\phi]X_i = F_i$$

$$\text{for } i = 1..M - 1 \quad (5)$$

Step 8. Convert each vector F_i into 2 dimensional eigen image as shown below in fig. 4.

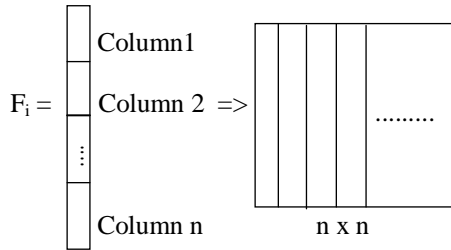


Fig.4. Conversion of a vector to an image

Step 9: Calculate the cumulative energy μ for each F_i image.

Step 10: Calculate the feature vector V_i for each training image I_i of the class as in equation 6

$$V_i = \begin{bmatrix} w_{1i} \\ w_{2i} \\ \vdots \\ w_{(M-1)i} \end{bmatrix} \quad (6)$$

$$\text{for } i = 1..M$$

where the coefficient w_{ji} is calculated as given in equation 7

$$w_{ji} = \frac{1}{\mu_j} \sum_{x,y} I_i(x,y) F_j(x,y) \quad (7)$$

$$\text{for } j = 1..M - 1, i = 1..M$$

After applying this entire procedure for all 'N' classes, we get the average image of each class, M-1 eigen images for each class and one feature vector of size (M-1)x1 for each training image of that class. Since total training images are MxN, we get 'MN' training feature vectors.

B. Feature vector generation for testing images

Each testing image is converted into gray scale image. The procedure used to generate feature vector for testing image I_{test} is given below(step 1 and step 2). This procedure is repeated for all other testing images.

For each training class do

Step1: Find zero mean test image as given in equation 8.

$$I_{ztest} = I_{test} - I_{avg} \quad (8)$$

Where I_{avg} is the average image of that class.

Step2: Calculate the feature vector V_{test} (refer equation 9)

$$V_{test} = \begin{bmatrix} w_{test1} \\ w_{test2} \\ \vdots \\ w_{testM-1} \end{bmatrix} \quad (9)$$

where each coefficient w_{testj} is given in equation 10.

$$w_{testj} = \frac{1}{\mu_j} \sum_{x,y} I_{ztest}(x,y) F_j(x,y) \quad (10)$$

Since there are 'N' classes, we get 'N' feature vectors for single test image. Each feature vector size is (M-1)x1.

C. Classification of testing image

When we apply the algorithms explained in section A, we get training feature vector set V containing MxN column feature vectors. Column vector V_{ij} denotes the feature vector for i^{th} training image of j^{th} class.

After applying the algorithm from section B we get feature vector set V_{test} (for single testing image) containing N column vectors. Each vector V_{testj} denotes the feature vector of testing image on j^{th} class. Procedure to classify the given testing image is given below:

```

For j=1 to N(Number of classes)
    For i=1 to M (Number of training images)
        D(i,j)=Euclidean_dist( $V_{ij}, V_{testj}$ )
    End
End
For j=1 to N
     $d_{min}(j)$ = Minimum value in  $j^{\text{th}}$  column of matrix D
End
J = Index of the minimum value of vector  $d_{min}$ 
    
```

The given testing image is assigned to J^{th} class. This procedure is executed for all testing images. Manhattan distance criteria are also used to find the distance between training and testing feature vector.

IV. RESULTS

The implementation of the proposed technique is done in MATLAB 7.0 using a computer with Intel Core i5, CPU (2.50GHz and 6 GB RAM). The proposed technique is tested on the COIL-100[21] image database. Columbia Object Image Library (COIL 100) is a database of color images of 100 objects. The objects have a wide variety of complex geometric and reflectance characteristics.

The objects were placed on a motorized turntable against a black background. The turntable was rotated through 360° to

vary object pose with respect to a fixed color camera. Images of the objects were taken at pose intervals of 5 degrees. This corresponds to 72 poses per object. Thus this database has total 7200 images. Fig. 5 shows the sample images of coil database objects. Fig. 6 shows the sample of different images in a particular class.



Fig.5. Sample images of objects in coil database



Fig.6. Sample images in one class of coil database

Initially for training, 10 images per class are used. These images are orderly selected from database at an angle of 35°,70°,105°,140°,.....so on. Remaining 62 images per class are used for testing purpose. Thus, out of 7200 images in database, 1000 images are used for training and 6200 images are used for testing purpose. Two distance criteria such as Euclidean distance (ED) and Manhattan distance (MD) are used for finding the difference between feature vectors. Results obtained are given in table I. Then the number of training images increased from 10 to 18. These images are orderly selected from database at an angle 0°,20°,40°,60°,.....so on. Remaining 54 images per class are used for testing purpose. Thus out of 7200 images in database, 1800 images are used for training and 5400 images are used for testing purpose. Table II shows the performance in the case.

TABLE I. NUMBER OF CORRECTLY CLASSIFIED IMAGES (OUT OF 62 PER CLASS, 100 CLASSES) (1000 TRAINING IMAGES AND 6200 TESTING IMAGES)

Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD
1	19	26	21	36	35	41	33	31	61	17	18	81	28	28
2	34	34	22	16	21	42	30	31	62	12	13	82	28	30
3	35	34	23	34	35	43	23	25	63	28	32	83	25	32
4	42	38	24	6	10	44	34	32	64	18	19	84	28	32
5	27	28	25	26	23	45	26	29	65	12	13	85	40	38
6	27	27	26	21	26	46	33	31	66	11	13	86	9	11
7	24	10	27	26	27	47	31	28	67	11	19	87	35	37
8	28	25	28	40	43	48	37	37	68	31	32	88	29	28
9	33	31	29	29	35	49	17	22	69	22	21	89	26	36
10	38	44	30	11	9	50	9	9	70	13	9	90	28	31
11	30	35	31	15	20	51	25	24	71	20	24	91	28	29
12	27	33	32	36	35	52	41	40	72	34	32	92	23	21
13	20	18	33	21	24	53	36	35	73	32	32	93	19	18
14	31	30	34	30	31	54	22	27	74	41	42	94	12	24
15	31	32	35	32	32	55	21	24	75	37	36	95	23	21
16	26	26	36	19	21	56	43	37	76	22	26	96	23	26
17	41	39	37	23	24	57	40	39	77	39	35	97	34	36
18	13	16	38	28	30	58	12	14	78	31	31	98	26	26
19	28	27	39	23	22	59	19	24	79	23	25	99	20	25
20	39	38	40	44	42	60	42	42	80	41	40	100	38	41

Note : Numbers in green indicate Euclidean distance gives better performance than Manhattan distance for that class. Numbers in yellow indicate Manhattan distance gives better performance comparatively for that class. No color indicates identical performance.

Observations: Out of 100 object classes, in 36 object classes, Euclidean distance gives better performance. In 53 object classes Manhattan distance outperforms Euclidean distance and in 11 object classes, both the distance criteria equally performed.

TABLE II. NUMBER OF CORRECTLY CLASSIFIED IMAGES (OUT OF 54 PER CLASS, 100 CLASSES) (1800 TRAINING IMAGES AND 5400 TESTING IMAGES)

Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD
1	29	43	21	41	42	41	41	40	61	32	33	81	42	48
2	31	42	22	24	27	42	45	43	62	41	41	82	46	45
3	48	46	23	46	47	43	44	41	63	49	43	83	37	34
4	44	36	24	22	25	44	44	48	64	28	28	84	23	25
5	33	29	25	27	30	45	38	45	65	22	29	85	46	47
6	37	34	26	33	35	46	42	41	66	29	28	86	19	14
7	34	22	27	44	44	47	39	34	67	20	33	87	24	33
8	36	33	28	48	48	48	40	40	68	42	42	88	39	41
9	38	37	29	37	43	49	28	32	69	27	29	89	38	46
10	42	44	30	16	12	50	24	22	70	13	13	90	39	43
11	38	38	31	31	38	51	36	37	71	39	40	91	35	32
12	30	31	32	32	26	52	46	44	72	36	38	92	25	27
13	29	32	33	37	38	53	45	44	73	43	44	93	47	47
14	37	43	34	48	41	54	37	34	74	47	48	94	22	29
15	38	38	35	26	31	55	36	37	75	47	48	95	0	0
16	39	40	36	44	47	56	31	31	76	34	34	96	35	37
17	38	37	37	40	41	57	46	45	77	40	41	97	39	42
18	23	23	38	44	47	58	28	27	78	44	44	98	25	29
19	45	45	39	32	34	59	29	30	79	42	44	99	32	33
20	43	41	40	51	48	60	44	42	80	49	47	100	47	35

Note : Numbers in green indicate Euclidean distance gives better performance than Manhattan distance for that class. Numbers in yellow indicate Manhattan distance gives better performance comparatively for that class. No color indicates identical performance.

Observations: Out of 100 object classes, in 33 object classes, Euclidean distance gives better performance. In 51 object classes Manhattan distance outperforms Euclidean distance and in 16 object classes, both the distance criteria equally performed.

Then the number of training images increased from 18 to 24. These images are orderly selected from database at an angle 0°,15°,30°,45°,.....so on. Remaining 48 images per class are used for testing purpose. Thus out of 7200 images in

database, 2400 images are used for training and 4800 images are used for testing purpose. Table III shows the performance in the case. Table IV shows the overall performance.

TABLE III. NUMBER OF CORRECTLY CLASSIFIED IMAGES (OUT OF 48 PER CLASS, 100 CLASSES) (2400 TRAINING IMAGES AND 4800 TESTING IMAGES)

Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD	Coil database Object number	ED	MD
1	33	36	21	47	47	41	45	43	61	27	20	81	44	44
2	41	41	22	23	25	42	48	46	62	31	32	82	45	42
3	48	45	23	48	47	43	48	46	63	47	47	83	40	29
4	46	41	24	9	10	44	47	47	64	19	12	84	25	27
5	15	14	25	18	23	45	34	44	65	25	27	85	48	47
6	45	44	26	37	37	46	42	43	66	27	24	86	39	30
7	2	0	27	44	41	47	28	22	67	24	29	87	22	30
8	42	43	28	48	48	48	46	47	68	47	46	88	38	35
9	36	37	29	30	25	49	38	37	69	34	35	89	41	45
10	43	42	30	18	17	50	15	15	70	19	16	90	45	44
11	38	35	31	42	34	51	40	39	71	34	34	91	39	39
12	28	24	32	24	17	52	48	48	72	44	43	92	32	30
13	22	23	33	42	45	53	48	47	73	46	46	93	30	30
14	46	46	34	42	39	54	40	41	74	47	47	94	11	15
15	47	48	35	29	34	55	41	37	75	45	45	95	10	8
16	43	41	36	46	46	56	31	37	76	38	39	96	37	38
17	46	47	37	47	47	57	46	46	77	44	42	97	45	46
18	26	24	38	48	48	58	33	28	78	48	48	98	23	29
19	46	48	39	41	40	59	32	33	79	41	41	99	31	30
20	47	45	40	48	48	60	48	46	80	48	48	100	46	47

Note : Numbers in green indicate Euclidean distance gives better performance than Manhattan distance for that class. Numbers in yellow indicate Manhattan distance gives better performance comparatively for that class. No color indicates identical performance.

Observations : Out of 100 object classes, in 45 object classes, Euclidean distance gives better performance. In 31 object classes Manhattan distance outperforms Euclidean distance and in 24 object classes, both the distance criteria equally performed.

TABLE IV. NUMBER OF CORRECTLY CLASSIFIED IMAGES AND ACCURACY

Distance criteria	Training images : 10 per class		Training images : 18 per class		Training images : 24 per class	
	Number of Correctly classified images (out of 6200)	Accuracy	Number of Correctly classified images (out of 5400)	Accuracy	Number of Correctly classified images (out of 4800)	Accuracy
Euclidean distance (ED)	2700	43.55	3602	66.70%	3695	76.98%
Manhattan distance (MD)	2799	45.15	3699	67.94%	3640	75.83%

Observations: In the first case, only 10 images per class are used for training and 62 images per class are used for testing, i.e. around 14% of data is used for training, so the accuracy is below 50%. In the second case, 18 images per class are used for training and 54 images per class are used for testing, i.e. 25% of data is used for training, so the accuracy is now increased above 65%. Manhattan distance criteria gives better overall accuracy comparatively in both the cases. In the third case when out of 72 images per class, 24 images per class

are used for training and 48 images per class are used for testing, i.e. around 33% of data is used for training, the accuracy is increased to above 75%. Euclidean distance criteria give better overall accuracy in comparison.

For different training database size, the accuracy for each class is calculated. Fig. 3,4 and 5 shows the number of classes in the different ranges of accuracy for different sizes of training databases.

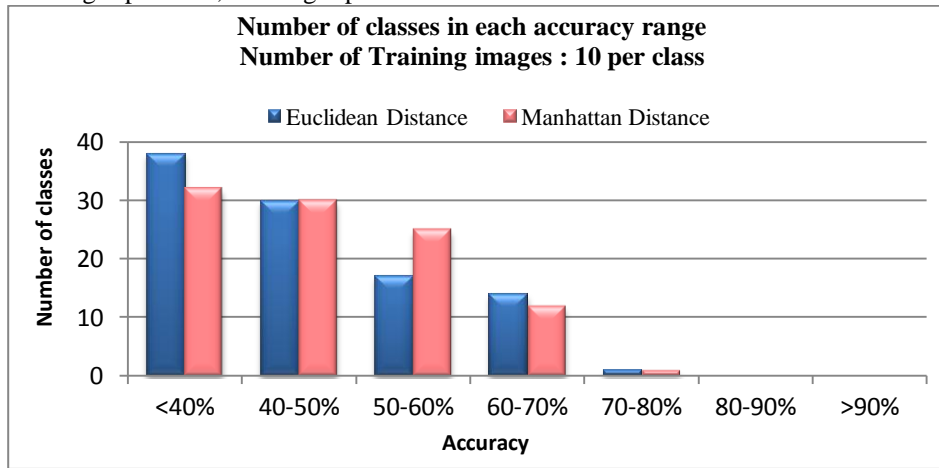


Fig.7. Accuracy performance of 100 classes when 1000 training images are used

Observations: When around 14% of data is used for training purpose, no object class gives more than 80% accuracy. Only one object class gives accuracy of 71%. Most of object classes give accuracy below 60%.

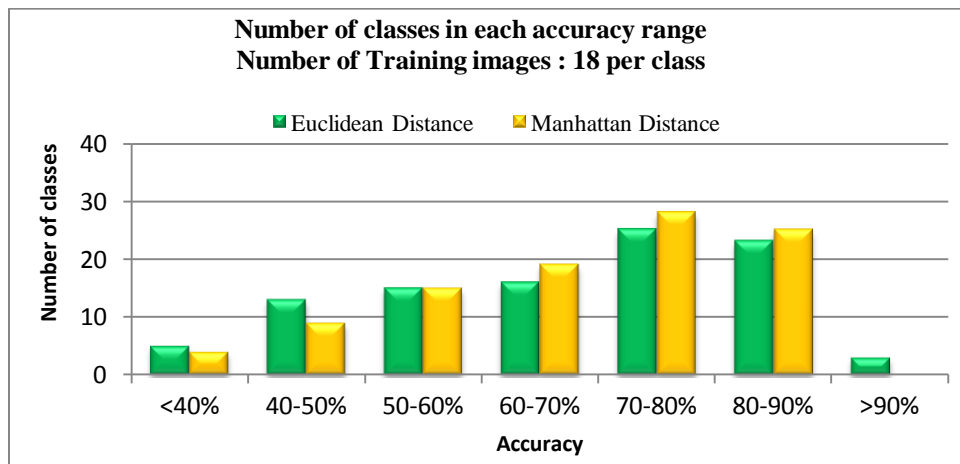


Fig.8. Accuracy performance of 100 classes when 1800 training images are used.

Observations: When 25% of data is used for training purpose, most of object classes give accuracy more than 60%. 51 object classes with Euclidean distance and 53 object classes with Manhattan distance give more than 70% accuracy. 3 classes give more than 90% accuracy with Euclidean distance.

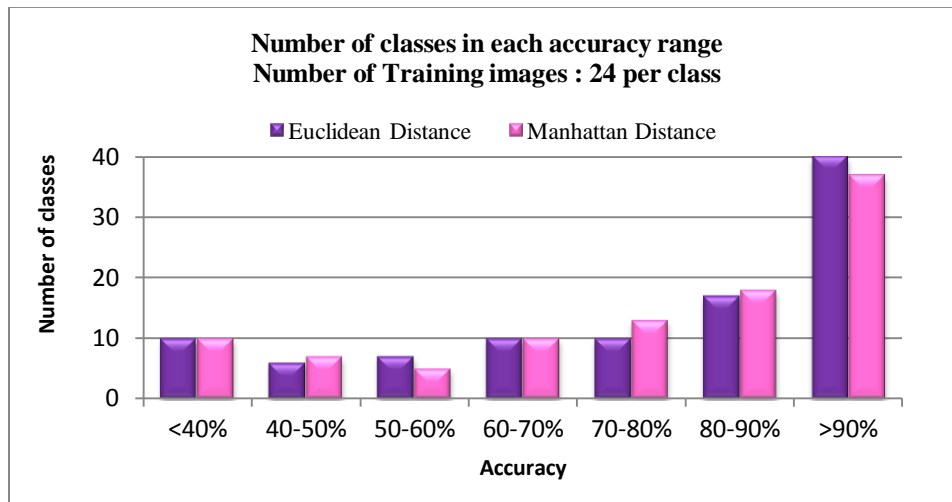


Fig.9. Accuracy performance of 100 classes when 2400 training images are used.

Observations: When around 33% of data is used for training purpose, most of object classes give more than 70% accuracy and around 40 classes give more than 90% accuracy. With Euclidean distance, 57 classes and with Manhattan distance, 55 classes give more than 80% accuracy.

V. CONCLUSIONS

The paper presents the application of PCA for an automatic classification of image database. Database used is COIL-100. It is a very large database containing 100 classes, each of 72 images, so total 7200 images. Each image is of size 128x128. If PCA is directly applied to an image for dimension reduction, then it would be computationally very intensive to find the eigen vectors of covariance matrix of size 128x128. In classification of data, generally training data is organized as columns of matrix and then PCA is applied to that matrix. But with such a large database, even if only 10 images per class are used for training purpose, the size of covariance matrix becomes 1000x1000. So to reduce the computational complexity, independent PCA is proposed and tested. In this technique the size of covariance matrix is $n \times n$ if 'n' images per class are used for training purpose. Experiments performed with three sizes of training database such as 10 images per class (13.88%), 18 images per class (25%) and 24 images per class (33.33%). When the training database is increased from around 14% to 25% and then to around 33%, the overall classification accuracy increases from 45% to 68% to 76%. Manhattan distance criterion gives overall better performance in comparison with Euclidean distance criterion when the size of training database is small. When 24 images per class are used for training, 40 object classes with Euclidean distance and 37 object classes with Manhattan distance give more than 90% accuracy. In this paper the technique is applied on grayscale image. It can be extended to all three planes of color image and combine the results.

References

- [1] H. Abdi and L.J. Williams, "Principal component analysis," Wiley Interdisciplinary Reviews: Computational Statistics, Vol.2, Issue 4, July 2010
- [2] K.Pearson, "On Lines and Planes of Closest Fit to Systems of Points in Space," Philosophical Magazine 2 (11),1901,pp. 559-572.
- [3] H.Hotelling,"Analysis of a complex of statistical variables into principal components," Journal of Educational Psychology, Vol 24(6), Sep 1933, 417-441.
- [4] H.B.Kekre, T.K.Sarode and J.K.Save, "Image Classification in Transform Domain," International Journal of Computer Science and Information Security (IJCSIS), Vol.10,No.3, Pages 91-97, Mar 2012.
- [5] H.B.Kekre, T.K.Sarode and J.K.Save, " Performance Comparison of Wavelet Transforms, Generated from Orthogonal Transforms, in Classification of Image Database," International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), Volume 2, Issue 12, December 2012 ISSN: 2277 128X, pp. 10-18
- [6] I.T.Jolliffe," Principal Component Analysis," Springer series in statistic, 2nd edition, New York:Springer; 2002.
- [7] S.K.Dandpat and S.Meher, " Performance Improvement for Face Recognition Using PCA and Two-Dimensional PCA, " IEEE International Conference on Computer Communication and Informatics (ICCCI), Jan. 2013, pp. 1-5
- [8] H. M. Hasan, Waleed A. ALJouhar and M. A. Alwan, "Face Recognition Using Improved FFT Based Radon by PSO and PCA Techniques, " International Journal of Image Processing (IJIP), Vol. 6, Issue : 1, 2012, pp. 26-37
- [9] H.B.Kekre, S. D. Thepade, and A. Maloo," Eigenvectors of Covariance Matrix using Row Mean and Column Mean Sequences for Face Recognition, " International Journal of Biometrics and Bioinformatics (IJBB), Vol. 4, Issue :2, pp. 42-50
- [10] H. B. Kekre, and K.Shah," Performance Comparison of Kekre's Transform with PCA and Other Conventional Orthogonal Transforms for Face Recognition, " Proceedings of the Second International Conference on Emerging Trends in Engineering and Technology (ICETET), 2009, pp. 873-879

- [11] H.Li and L.Wang, "Palmpoint Recognition Using Dual-Tree Complex Wavelet Transform and Compressed Sensing , " IEEE International Conference on Measurement, Information and Control (MIC), May 2012,Vol.2, pp.563-567
- [12] Mohammad Bonab and Mostafa Bonab, "A New Technique for Image Compression Using PCA , " International Journal of Computer Science & Communication Networks,Mar. 2012,Vol. 2, No. 1, pp. 111-116
- [13] V.Gaidhane, V.Singh and M.Kumar, "Image Compression Using PCA and Improved Technique with MLP Neural Network, " IEEE International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom), Oct 2010, pp. 106-110
- [14] Nirosha Joshitha J, R. Medona Selin, "Image Fusion using PCA in Multifeature Based Palmpoint Recognition, " International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Vol. 2, Issue : 2, May 2012, pp. 226-230
- [15] Qiang Guo and Ya Chen, "Image Resolution Enhancement Using PCA Based Post Filtering, " Research Journal of Applied Sciences, Engineering and Technology, Maxwell Scientific Organization ,Vol.5, No. 2, Jan 2013, pp. 602-605
- [16] Khan, M.A., Khan, A., Mahmood, T., Abbas, M. and Muhammad, N., "Fingerprint image enhancement using Principal Component Analysis (PCA) filters, " IEEE International Conference on Information and Emerging Technologies (ICIET), 978-1-4244-8003-6/10/June 2010, pp.1-6
- [17] M. Asunción Vicente, Cesar Fernández, Oscar Reinoso, and Luis Payá, "3D Object Recognition from Appearance: PCA Versus ICA Approaches, " Image Analysis and Recognition Lecture Notes in Computer Science , Springer Publication, Volume 3211, 2004, pp 547-555
- [18] L. C. PaulI, A. A. Suman, N. Sultan, "Methodological Analysis of Principal Component Analysis (PCA) Method," International Journal of Computational Engineering & Management, IJCEM, Vol. 16 Issue 2, March 2013, ISSN (Online): 2230-7893,pp. 32-38
- [19] I. S.Bajwa, and S.I.Hyder, "PCA based Image Classification of Single-layered Cloud Types," IEEE -2005 International Conference on Emerging Technologies, September 17-18, Islamabad, pp. 365-369
- [20] Smith, Lindsay I. "A tutorial on principal components analysis." Cornell University, USA 51 (2002): 52.
- [21] Nene S. A., Nayar S. K. and Murase H.: Columbia Object Image Library (COIL-100), Technical Report CUCS-006-96 (1996)

AUTHORS PROFILE



Dr. H. B. Kekre has received B.E. (Hons.) in Telecomm. Engineering. from Jabalpur University in 1958, M.Tech (Industrial Electronics) from IIT Bombay in 1960, M.S.Engg. (Electrical Engg.) from University of Ottawa in 1965 and Ph.D. (System Identification) from IIT Bombay in 1970 He has worked as Faculty of Electrical Engineering and then HOD Computer Science and Engg. at IIT Bombay. For 13 years he was working as a professor and head in the Department of Computer Engg. at Thadomal Shahani Engineering College, Mumbai. Now he is Senior Professor at MPSTME, SVKM's NMIMS University. He has guided 17 Ph.Ds, more than 100 M.E./M.Tech and several B.E./ B.Tech projects. His areas of interest are Digital Signal processing, Image Processing and Computer Networking. He has more than 450 papers in National /International Conferences and Journals to his credit. He was Senior Member of IEEE. Presently He is Fellow of IETE and Life Member of ISTE Recently twelve students working under his guidance have received best paper awards and six research scholars have been conferred Ph. D. Degree by NMIMS University. Currently 7 research scholars are pursuing Ph.D. program under his guidance.



Tanuja K. Sarode has Received Bsc. (Mathematics) from Mumbai University in 1996, Bsc.Tech.(Computer Technology) from Mumbai University in 1999, M.E. (Computer Engineering) from Mumbai University in 2004, currently Pursuing Ph.D. from Mukesh Patel School of Technology, Management and Engineering, SVKM's NMIMS University, Vile-Parle (W), Mumbai, INDIA. She has more than 10 years of experience in teaching. Currently working as Associate Professor in Dept. of Computer Engineering at Thadomal Shahani Engineering College, Mumbai. She is life member of IETE, ISTE, member of International Association of Engineers (IAENG) and International Association of Computer Science and Information Technology (IACSIT), Singapore. Her areas of interest are Image Processing, Signal Processing and Computer Graphics. She has more than 100 papers in National /International Conferences/journal to her credit.



Jagruti K. Save has received B.E. (Computer Engg.) from Mumbai University in 1996, M.E. (Computer Engineering) from Mumbai University in 2004, currently Pursuing Ph.D. from Mukesh Patel School of Technology, Management and Engineering, SVKM's NMIMS University, Vile-Parle (W), Mumbai, INDIA. She has more than 10 years of experience in teaching. Currently working as Associate Professor in Dept. of Computer Engineering at Fr. Conceicao Rodrigues College of Engg., Bandra, Mumbai. Her areas of interest are Image Processing, Neural Networks, Fuzzy systems, Data base management and Computer Vision. She has 8 papers in National /International Conferences/journal to her credit.