Journal of Computer Science 2 (1): 53-58, 2006
ISSN 1549-3636

Splitting Technique Initialization in Local PCA

[1]Alok Sharma, [1]Kuldip K. Paliwal and [2]Godfrey C. Onwubolu
[1]School of Microelectronics Engineering, Griffith University, Brisbane, Australia
[2]Department of Engineering, University of the South Pacific, Suva, Fiji

Abstract: The local Principal Component Analysis (PCA) reduces linearly redundant components that may present in higher dimensional space. It deploys an *initial guess technique* which can be utilized when the distribution of a given multivariate data is known to the user. The problem in initialization arises when the distribution is not known. This study explores a technique that can be easily integrated in the local PCA design and is efficient even when the given statistical distribution is unknown. The initialization using this proposed *splitting technique* not only splits and reproduces the mean vector but also the orientation of components in the subspace domain. This would ensure that all clusters are used in the design. The proposed integration with the reconstruction distance local PCA design enables easier data processing and more accurate representation of multivariate data. A comparative approach is undertaken to demonstrate the greater effectiveness of the proposed approach in terms of percentage error.

Key words: Local PCA, hybrid distance, vector quantization, splitting technique, VQPCA

INTRODUCTION

Dimension reduction methods are employed in statistical pattern classification problem to represent higher dimensional embeddings in a lower dimensional space by eliminating or removing the redundant components that may present in multivariate data so that the data loss is minimal. The interpretation of multivariate data or feature vectors becomes quite unmanageable when the dimension size is high. This severely increases the memory/storage requirements and augments the problems in pattern classification. It then becomes essential to express and understand a given high dimensional vector onto a parsimonious data space that best describes the feature vectors. The given multivariate data can be in the form of image, sound/speech, financial data or any statistical data. The given data depend upon several characteristics; for example, in face recognition, the classification of faces depends upon the location of eyes, width and height of nose/mouth, length of eyebrows, complexion etc. All these characters constitute as one vector of a given multivariate data.

The conventional technique for dimension reduction is PCA also known as Karhunen-Loéve technique (KLT)[1]. The objective of PCA is to reduce redundant dimensional components subject to minimal loss of information. It will find a global linear transform of a given data in the feature space. The linear transform gives several basis vectors. The first basis vector will be in the direction of maximum variance of the given data. The second basis vector will be mutually orthogonal to the first basis vector. Similarly the remaining basis vectors are mutually orthogonal to the previous basis vectors and in order, maximize the remaining variances subject to the orthogonal condition. The principal axes of PCA are those orthonormal axes onto which the remaining variances under projection are maximum. These orthonormal axes are given by the dominant eigenvectors i.e. those eigenvectors that corresponds to the largest associated eigenvalues. The obtained components in reduced dimensional space are optimal in minimum mean square error (MSE) sense.

Perceiving the constraints involved in PCA, researchers have extended the basic PCA model. Oja[2] introduced a simple linear neuron model for PCA with constrained Hebbian-type modification and derived unconstrained learning rules and showed how the neuron model extracts the one dimensional principal components. Several other neural network algorithms for PCA have also been developed[3-9]. Hastie[10] introduced principal curves and surfaces as the estimates of non-linear generalizations of linear one dimensional PCA technique and Tibshirani[11] presented an alternative definition of principal curves based on a mixture model. Tipping and Bishop[12] demonstrated how principal axes of a set of observed patterns are determined through maximum likelihood estimation. Xu[13], De la Torre and Black[14] and Koren and Carnel[15] suggested robust PCA model which can perform well under the presence of outliers. A non-linear form of PCA[16], local linear PCA[17-19] and mixture of local PCA[20] have also been developed. In the local linear PCA approach a class is partitioned into several disjoint regions by vector quantization and then

Corresponding Author: Alok Sharma, School of Microelectronics Engineering, Griffith University, Brisbane, Australia, Tel: +679 321 2870/ +617 3875 3754, Fax: +679 323 1538

performs local PCA about each cluster. This local PCA approach is further extended by utilizing hybrid distance measure also referred to as reconstruction distance[17], which has been proved to be one of the optimal techniques in terms of producing low reconstruction error. Local PCA based on hybrid distance is a replacement of Euclidean distance, which has been proved to be a better distance measurement tool in local linear approach for the cluster separability[17] (Vector quantization is one of the approaches used in removal of redundant feature vectors subject to minimal loss of information. This technique separates a set of feature vectors into small and dense regions known as Voronoi regions[21] and estimates the feature vectors in the obtained regions by their corresponding mean, referred as codewords. The number of Voronoi regions in a given class is known as the levels of the quantizer. For further information on VQ refer[22,23]). This distance criteria is derived from the MSE of the system. The hybrid-distance local PCA also known as vector quantization principal component analysis (VQPCA) for reconstruction distance deploys an *initial guess technique*[23]. This *initial guess technique* is usually applied to the data when there is at least some information about the distribution since an arbitrary initialization could lead to poor performance. This means that for a known statistical distribution (not completely) the reproductive vectors or codewords are initially defined by manually placing them in the vicinity of the data. The number of codewords previously defined will not change during the process except the location of codewords, which will be updated until the best region or cluster is found. In most of the practical cases, statistical distribution is not known to the user which augments the problem of initialization of codewords. Furthermore, in VQPCA some of the codewords if not very carefully selected, end up being isolated having no samples associated to it. This restricts the performance of VQPCA and thus the model is strongly dependent on the selection of initial codewords.

For VQ algorithms alone, several extensions have been developed[24-29] to improve the performance and overcome drawbacks. Whereas in VQPCA direct implementation of splitting Linde-Buzo-Gray (LBG) technique[23] cannot be integrated with the hybrid-distance local PCA design since it does not accounts for updating and reproducing eigenvectors (directional vector) with the corresponding covariance matrix of the local regions. To overcome this type of problems associated with the hybrid-distance local PCA technique, we have presented a *splitting* initialization approach that can be easily integrated in local PCA design and is efficient even when the given statistical distribution is unknown. The introduced approach not only updates centroid (mean) of a cluster but also the orientation of components in subspace domain with the corresponding covariance matrix, through splitting and reproducing code words. Overall one can view this proposed approach as an improved hybrid-distance local PCA which can efficiently accommodate processing and clustering of unknown statistical distributed data. For brevity we refer to this approach as VQPCA-sp in this study, where the suffix *sp* denotes initialization of VQPCA using *splitting technique*.

DESIGN MODEL

Here, it is elaborated that the VQPCA-sp approach using hybrid-distance as a distance measurement tool. To explain hybrid-distance, suppose in a d-dimensional hyperplane, $\underline{\mu}_{im}$ denotes the mean vector of m^{th} cluster in i^{th} discrete block (class) and $\underline{\phi}_{k}^{im}$ denotes k^{th} eigenvector of m^{th} cluster which is in i^{th} class, then hybrid-distance is defined as:

$$dist(x,\mu,\phi) = (\underline{x} - \underline{\mu}_{im})^{t}\, \underline{\underline{P}}_{im} (\underline{x} - \underline{\mu}_{im}) \qquad (1)$$

Where, $\underline{\underline{P}}_{im}$ is a local projection matrix which projects the feature vectors onto a subspace orthogonal to the local h-dimensional PCA hyperplane[17] i.e.

$$\underline{\underline{P}}_{im} = \sum_{k=h+1}^{d} (\underline{\phi}_{k}^{im})(\underline{\phi}_{k}^{im})^{t} \quad \text{where } h < d \qquad (2)$$

It is evident from the expression of the hybrid-distance (equations 1 and 2) that it depends upon eigenvector and mean of the local clusters. Thereby these two vectors should be taken under consideration for the reproduction.

In VQPCA-sp approach firstly, the set of feature vectors is separated into disjoint regions or clusters by applying vector quantization technique for each given class(There is a fundamental difference between a class and a cluster, class represents a set of feature vectors or parameters of a distinct element which can accommodate several clusters i.e. a cluster is a subset of a class. For example the feature vectors of vowels /a/ and /e/ will form two distinct classes. In either of the class there could be several small partitions referred as clusters) and then local PCA is performed using *splitting* technique about each of the cluster center using hybrid-distance. In other words this approach segregates data by class and then performs VQPCA on each class using *splitting* technique. If Euclidean-distance is used in place of hybrid-distance then LBG algorithm could simply be integrated for local PCA implementation. In this case, the regions or clusters are partitioned independently without considering the orientation of PCA and thus produces suboptimal results. On the other hand, the hybrid-distance as given in equation 1 depends not only upon the mean of the clusters but also upon the eigenvectors of the covariance matrix of the clusters. In this case a direct splitting LBG algorithm or Enhance LBG algorithm[29] cannot be integrated in the local PCA design since an iterative process is required to reproduce eigenvectors as well by updating covariance matrix of the clusters. An *initial guess technique* was utilized[17] for hybrid-distance local PCA design where codewords are initialized to random input vectors from the training dataset. If poorly initialized this initialization approach may lead some clusters not to be used at all. If a user has some prior knowledge about the statistical distribution of a given feature vectors then the performance (in terms of percentage reconstruction error) could improve further. However, this *a priori* information is not always present when the feature vectors are given for processing. In many pragmatic cases, *initial guess technique* is not appropriate and thus it requires some technique that does not require prior knowledge of data distribution. On the other hand *splitting* technique approach does not require any *a priori* information of the data distribution because the initialized codeword is the center of the data. Thereafter it splits and searches for the region for which the expected error is minimum. Figure 1 depicts *splitting* approach on a given two dimensional data presented in an elliptical form. Firstly the initial mean μ^0 and initial eigenvector, ϕ_1^0 as primary component and ϕ_2^0 as secondary component are computed respectively. These reproductive vectors are perturbed using small predefined quantity to get a slight variation in the values. In the figure two new mean vectors after splitting are μ^+ and μ^- and their corresponding local eigenvectors are defined as $\Phi^+ = (\phi_1^+, \phi_2^+)$ and $\Phi^- = (\phi_1^-, \phi_2^-)$ respectively. Any arbitrary vector *P* of the feature vectors (Fig. 1) is taken into account to determine the membership of the vector to one of the two separated regions (either Φ^+ or Φ^-) as defined by their reproductive vectors. This will form two new clusters and the mean and eigenvector will be updated. The reproduction of the two set of vectors will be carried iteratively until the distortion in reconstruction is smaller than some threshold error (predefined value). Once the satisfactory distortion level is achieved, the reproductive vectors split again and carry out the same above iterative process, until the desired number of clusters is obtained. The determination of membership of the arbitrary point is done by using hybrid-distance. The VQPCA-sp accommodates both of the vectors (mean and eigenvector) by updating and reproducing them using iterative process which will be discussed later. The improved hybrid-distance local PCA could be of two types (i) where splitting occurs at random without following any particular direction as illustrated in Fig. 1 and (ii) where split follows the direction of dominating or principal component.

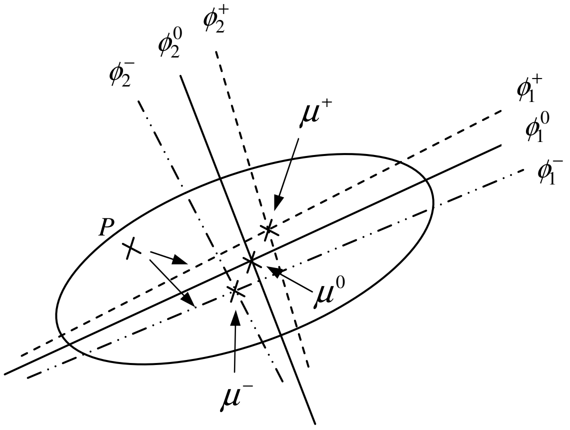


Fig. 1: Splitting initialization process

The principal component refers to the eigenvector for which the corresponding eigenvalue is maximum.

IMPLEMENTATION SCHEME

It deals with the implementation scheme of VQPCA-sp using hybrid-distance as a prototype. Suppose in a *c*-class problem a set of class is defined by $W = \{\omega_i; i = 1,2,...,c\}$ where label ω_i denotes i^{th} class; each class is subdivided into a set of clusters defined by $\xi^i = \{C_k^i; k = 1,2,...,N\}$, where $N > 2^p$ denotes the total number of desired clusters (levels) that is required for each given class and *p* is any real integer greater than or equal to one; C_k^i denotes k^{th} cluster in ω_i. Let *d*-dimensional training data in ω_i be defined as $x = \{\underline{x}_j^{(i)}; j = 1,2,...,n_i\}$ where n_i is the number of samples per given class, $\underline{x}_j^{(i)}$ denotes any arbitrary feature vector. The transformation $f : \Re^d \to \Re^h$ is from *d*-dimensional hyperplane to *h*-dimensional hyperplane/plane such that $h < d$. By considering all the mentioned terms above, the VQPCA-sp algorithm can be given as follows:

Step 0: Define threshold error $\varepsilon > 0$, initial average distortion $D_1 \to \infty$ for class ω_1.

Step 1: Initialize mean $\underline{\mu}$ and covariance $\underline{\underline{\Sigma}}$ as $\underline{\mu}_{i1} = \frac{1}{n_i} \sum_{x \in \omega_i} x$; $\underline{\underline{\Sigma}}_{i1} = \frac{1}{n_i} \sum_{x \in \omega_i} (x - \underline{\mu}_{i1})(x - \underline{\mu}_{i1})^t$ where $i = 1,2,...,c$. Set the variable level $M = 1$.

Step 2: Apply KLT to obtain $d \times d$ eigenvector set $\underline{\underline{\Phi}}_{im} = \{\underline{\phi}_l^{im}; l = 1,2,...,d\}$ arranged according to their corresponding eigenvalue set which is placed in descending order, where $\underline{\phi}_l^{im}$ is any arbitrary eigenvector of ω_i class and C_m^i level, for $m=1,2,...,M$. Split the reproductive vectors as $\underline{\mu}_{im}=[\underline{\mu}_{im}+\varepsilon,\underline{\mu}_{im}-\varepsilon]$ and $\underline{\underline{\Phi}}_{im}=[\underline{\underline{\Phi}}_{im}+\varepsilon,\underline{\underline{\Phi}}_{im}-\varepsilon]$ if the direction of splitting is allowed to be random, otherwise $\underline{\mu}_{im}=[\underline{\mu}_{im}+\varepsilon\underline{\phi}_1^{im},\underline{\mu}_{im}-\varepsilon\underline{\phi}_1^{im}]$ is used when the splitting is following the direction of principal component $\underline{\phi}_1^{im}$ for which the corresponding eigenvalue is maximum. Update level $M\leftarrow 2M$.

Step 3: Given the sum of $(d-h)$ trailing eigenvectors $\underline{\underline{P}}_{im}=\sum_{k=h+1}^{d}(\underline{\phi}_k^{im})(\underline{\phi}_k^{im})^t$, compute the reconstruction distance $d(x,\underline{\mu}_{im},\underline{\underline{P}}_{im})=(x-\underline{\mu}_{im})^t\underline{\underline{P}}_{im}(x-\underline{\mu}_{im})$ to get the minimum distance $\delta_{\min}=\min_{m=1,2,...,M}[d(x,\underline{\mu}_{im},\underline{\underline{P}}_{im})]$. Find all feature vectors $x\in C_k^i$ where $k=\arg[\delta_{\min}]$. For updating $\underline{\mu}$ and $\underline{\underline{\Sigma}}$ of the new partitioned Voronoi regions, equations $\underline{\mu}_{im}=\frac{1}{n_{i_m}}\sum_{x\in C_m^i}x$ and $\underline{\underline{\Sigma}}_{im}=\frac{1}{n_{i_m}}\sum_{x\in C_m^i}(x-\underline{\mu}_{im})(x-\underline{\mu}_{im})^t$ are used, where n_{i_m} represent number of samples in the new C_m^i Voronoi region. Iterate the process until $(D_{f-1}-D_f)/D_f\le\varepsilon$ where $D_f=\frac{1}{n_i.d}\sum_{x\in\omega_i}\delta_{\min}$ and f is some iterative number. Follow next step with the revised values of $\underline{\mu}_{im}$ and $\underline{\phi}_l^{im}$.

Step 4: Iterate step 2 and step 3 until M equalizes the value of N.

Step 5: Follow the same procedure (step 1 – step 4) for all the remaining classes $\{\omega_i; i=2,3,...,c\}$.

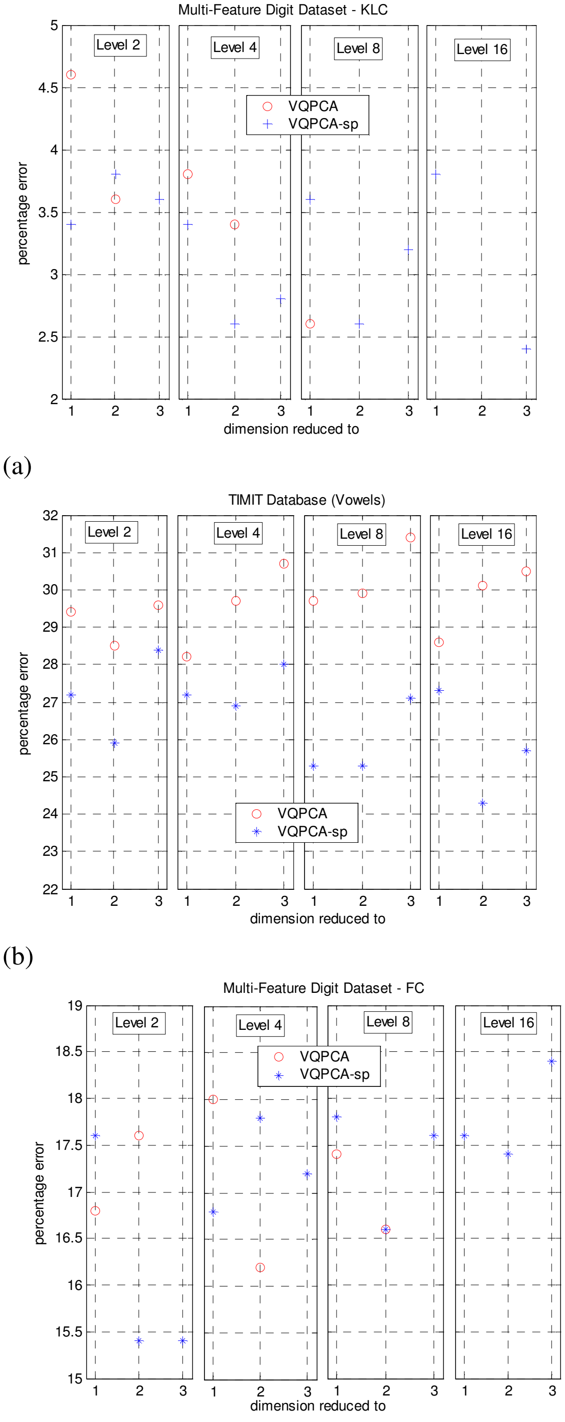
For the reconstruction of vector, expression

$$\hat{x}=\left(\sum_{l=1}^{h}\underline{\phi}_l^{im}(\underline{\phi}_l^{im})^t\right)(x-\underline{\mu}_{im})+\underline{\mu}_{im}$$

is used where $i=1,2,...,c$ and $m=1,2,...,N$. The normalized difference between the vector x and the reconstructed vector $\hat{x}$ is the reconstruction error.

RESULTS

Several machine learning corpuses have been employed for estimating the accuracy of the proposed model and the existing model. Figure 2 depicts percentage error as a function of dimension reduction at levels 2, 4, 8 and 16.



(a)

(b)

(c)

Fig. 2: Percentage error as a function of dimension reduction at levels 2, 4, 8 and 16 on machine learning corpuses

The percentage error obtained by VQPCA and VQPCA-sp are represented as small circles ('o') and asterisk sign ('*') respectively. It is observed that at some points no results were obtained; this is due to the fact that codewords are left out alone, having no feature vectors associated to them, which is an erroneous situation and has not been considered in the testing session. The reconstruction or hybrid distance has been used as a prototype for distance measurement in all the designs.

Figure 2a exhibits a design using Multi-Feature Digit Dataset[30,31] of 64 dimensional Karhunen-Loéve coefficients. A total of 1500 vectors of 10 distinct classes were utilized for training session and a total of 500 vectors were used for testing the system. For VQPCA model codewords are initialized close to the centre of samples and updated iteratively until the best cluster in terms of minimum mean square error is achieved. Dimension is reduced from 64 to 1, 2 and 3. It can be observed that VQPCA-sp model demonstrates better performance in all the selected levels in terms of producing lesser error and greatly overcoming the problem of codewords being isolated. For example at level-16 no results are obtained for VQPCA model, this depicts the strong dependence of VQPCA model on the initial selection of codewords, whereas VQPCA-sp produces 2.4% error.

Figure 2b illustrates a classifier design using 10 distinct vowels from TIMIT database[32]. A total of 6000 mel-frequency cepstral coefficients with energy-delta-acceleration (MFCC_E_D_A)[33] vectors of dimension 39 in training session and 2000 MFCC_E_D_A vectors in testing session were used. Dimension reduction is from 39 to 1, 2 and 3. Here again VQPCA-sp produces much better performance than VQPCA, producing up to 24.3% error whereas minimum error obtained by VQPCA is 28.2%.

In Fig. 2c a classification design using Multi-Feature Digit Dataset[30,31] of 76 dimensional Fourier coefficients was undertaken. A sum of 1500 vectors of 10 distinct classes was utilized to train the classifier. Then a separate set of 500 vectors was used for validation. Dimension is reduced from 76 to 1, 2 and 3. Here also VQPCA-sp exhibits better performance than VQPCA at almost all the levels. The lowest error noted by VQPCA-sp is 15.4% and that by VQPCA is 16.2%. At some points (level-4 dimension 3, level-8 dimension 3, level-16 dimension 1-3) VQPCA is not able to produce any result. However, VQPCA-sp produces result at all the points.

It could be observed from the experiments that VQPCA-sp method produces better representation of multivariate data and able to overcome up to the greater extent the problem related to codewords being isolated with no samples associated to it.

CONCLUSION

This study has described a new splitting technique on local PCA approach (VQPCA) utilizing hybrid-distance as a distance measure tool for cluster separation. It was observed from the experiments on several machine learning corpuses that the proposed approach produces more accurate representation of multivariate data in reduced dimensional space. This VQPCA-sp approach is efficient even when the given distribution of statistical data is unknown. It was also experienced that VQPCA-sp was much easier in initializing codewords and the probability of codewords being left alone was much less as compared to VQPCA. The percentage error obtained by VQPCA-sp model is independent of initial codeword selection since the codewords are selected involuntarily starting from the center of the considered data. This method not only splits mean but also the orientation of the components on a regular iterative basis which was not accommodated on VQPCA alone.

REFERENCES

1. Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition. Academic Press Inc., Hartcourt Brace Jovanovich, Publishers.
2. Oja, E., 1982. A Simplified Neuron Model as a Principal Component Analyzer. J. Math. Biology, 15: 267-273.
3. Baldi, P. and K. Hornik, 1995. Learning in linear neural networks: A survey. IEEE Trans. Neural Network, 6: 837-858.
4. Peper, F. and H. Noda, 1995. A class of simple nonlinear 1-unit PCA neural networks. IEEE Intl. Conf. Neural Networks, 1: 285-289.
5. Oliveira, P.R. and R.F. Romero, 1996. A comparison between PCA neural networks and the JPEG standard for performing image compression. Workshop on Cybernetic Vision, pp: 112-116.
6. Diamantaras, K.I., 1998. Asymmetric PCA neural networks for adaptive blind source separation. Neural Networks for Signal Processing VIII Proc. IEEE Signal Processing Society Worksop, pp: 103-112.
7. Wang, S., Y. Liang and F. Ma, 1998. An adaptive robust PCA neural network. IEEE Jnt. Conf. Neural Networks, 3: 2288-2293.
8. Tao, F., L. Jiangang, W. Zhi and S. Youxian, 2003. An image compressing algorithm based on PCA/SOFM hybrid neural network. Industrial Electronics Conf. (IECON '03), 3: 2103-2107.
9. Chin-Teng, L., C. Shi-An, H. Chao-Hui and C. Jen-Feng, 2004. Cellular neural networks and PCA neural networks based rotation/scale invariant texture classification. IEEE Intl. Jnt. Conf. Neural Networks, 1: 158.

10. Hastie, T., 1984. Principal curves and surfaces. Ph.D. Thesis. Stanford University, Calif.
11. Tibshirani, R., 1992. Principal curves revisited. Statistics and Computing, 2: 183-190.
12. Tipping, M.E. and C.M Bishop, 1998. Mixtures of probabilistic principal component analyzers. Tech. Rep. NCRG/97/003, Neural Computing Research Group, Aston University.
13. Xu, L., 1995. Robust principal component analysis by self-organizing rules based on statistical physics approach. IEEE Trans. Neural Networks, 6: 131-143.
14. De la Torre, F. and M.J. Black, 2001. Robust principal component analysis for computer vision. Intl. Conf. Computer Vision (ICCV), Vancouver, Canada, pp: 362-369.
15. Koren, Y. and L. Carnel, 2004. Robust linear dimensionality reduction. IEEE Trans. Visualization and Computer Graphics, 10: 459-470.
16. Schölkopf, B., A. Smola and K.R. Müller, 1998. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 10: 1299-1319.
17. Kambhatla, N. and T.K. Leen, 1997. Dimension Reduction by local PCA. Neural Computing, 9:1493-1516.
18. Dony, R.D. and S. Haykin, 1995. Optimally adaptive transform coding. IEEE Trans. Image Process., pp: 1358-1370.
19. Dony, R.D. and S. Haykin, 1997. Compression of SAR images using KLT, VQ and mixture of principal components. IEE Proc. Radar Sonar Navig., 144: 113-120.
20. Dony, R.D. and S. Haykin, 1997. Image segmentation using mixture of principal components representation. IEE Proc. Vis. Image Signal Process., 144: 73-80.
21. Makhoul, J., S. Roucos and H. Gish, 1985. Vector quantization in speech coding. IEEE Proc., 73: 1151-1588.
22. Gray, R.M., 1984. Vector Quantization. IEEE ASSP Mag., pp: 4-29.
23. Linde, Y., A. Buzo and R.M. Gray, 1980. An Algorithm for vector quantization design. IEEE Trans. Commun., 28: 84-94.
24. Gresho, A. and R. Gray, 1992. Vector Quantization and Signal Compression. Boston, Kluwer.
25. Paliwal, KK. and B.S. Atal, 1993. Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame. IEEE Trans. Speech and Audio Process., 1: 3-14.
26. Karayiannis, N., 1997. A Methodology for Constructing Fuzzy Algorithms for Learning Vector Quantization. IEEE Trans. on Neural Networks, 8: 505-518.
27. Hofmann, T. and J. Buhmann, 1997. Pairwise data clustering by deterministic annealing. IEEE Trans. Pattern Analysis and Machine Intelligence, 19: 1-14.
28. Fritzke, B., 1997. The LBG-U method for vector quantization–An improvement over LBG inspired from neural network. Neural Process. Lett., 5: 35-45.
29. Patané, G. and M. Russo, 2001. The enhanced LBG algorithm. Neural Networks, 14: 1219-1237.
30. Jain, A.K., R.P.W. Duin and J. Mao, 2000. Statistical pattern recognition: A review. IEEE Trans. Pattern Analysis and Machine Learning, 22: 4-37.
31. Blake, C.L. and C.J. Merz, 1998. UCI repository of machine learning databases. http://www.ics.uci.edu/~mlearn, Irvine, CA, University of California, Department of Information and Computer Science.
32. Garofolo, S.J, L.F. Lori, F.M. William, F.G. Jonathan, P.S. David and D.L. Nancy, 1986. The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CDROM, NIST.
33. Young, S., G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev and P. Woodland, 2002. The HTK Book Version 3.2, Cambridge, England, Cambridge University.