



ELSEVIER

Pattern Recognition Letters 17 (1996) 1131–1139

Pattern Recognition
Letters

Extended subspace methods of pattern recognition

M. Prakash, M. Narasimha Murty *

Department of Computer Science and Automation, Indian Institute of Science, Bangalore 560 012, India

Received 11 January 1996; revised 3 May 1996

Abstract

The Subspace Pattern Recognition Method (SPRM) is a statistical method where each class is represented by a separate subspace. There are a number of variants to it including the Averaged Learning Subspace Method (ALSM). The decision surfaces in all these methods are quadratic. In some applications, we may require decision surfaces which are more nonlinear in nature. In this paper, we have proposed the use of more than one subspace (cluster) in the representation of the classes in the subspace methods of pattern recognition. By keeping the number of principal components in all the clusters the same, this model allows for a piecewise linear approximation of the decision surfaces. We have used this model to extend both the SPRM and the ALSM to obtain the Extended SPRM and the Extended ALSM, respectively. We have investigated the use of a dynamic clustering algorithm for the assignment of patterns to different clusters as opposed to a random assignment. We have conducted experiments on three data sets including a 192-dimensional large character data set. The results indicate that the proposed methods have the potential to approximate any decision surface, and can considerably improve the classification accuracy on the test sets.

1. Introduction

Watanabe (1969) suggested the *Subspace Pattern Recognition Method* (SPRM) as a new approach to classifying and representing patterns given as elements of a vector space. Here, each class is represented by a subspace spanned by a group of basis vectors – the orthogonal components obtained by principal component analysis. One drawback with the basic SPRM is that the features obtained for one class are not dependent on the features of the other classes. The features chosen are very good for data compression, and thus are good at representing an average pattern. However, if there is a large number of *outlying samples* in the tails of the probability density functions for the classes, these features will no longer be good. Improvements

to the basic SPRM can be put under the following categories: (i) methods based on weighted orthogonal projections, and (ii) methods based on the rotation of subspaces.

There have been many attempts of associating weights to the basis vectors in the computation of the orthogonal projections (see Oja, 1983). All these methods obtain the weights a priori based on the first- and second-order statistical properties. Recently, learning methods to assign weights using the training samples have been proposed using a genetic algorithm (Prakash and Murty, 1995a) (here, the weights are binary) and a neural network based on the Hebbian Learning Subspace Method (HLSM) (Prakash and Murty, 1995b). By adapting the weights, these learning methods implicitly integrate feature extraction and classification.

The first improvement of the SPRM based on the

* Corresponding author. E-mail: mnm@csa.iisc.ernet.in.

rotation of subspaces is proposed by Kohonen et al. (1979). They suggested decision directed learning in their Learning Subspace Method (LSM) to improve the performance. The representations (i.e., the subspaces or the orthonormal features/principal components) get *adapted* in the process of rotation. Thus, the LSM integrates feature extraction and classification. A number of variants of this have been proposed (see Oja, 1983).

The learning methods have been successfully applied in practice in a variety of applications including speech recognition (Oja, 1983), texture identification (Oja and Parkkinen, 1986), and character recognition (Prakash and Murty, 1995b). In practice, these methods are fast and exhibit reasonably robust behavior with respect to convergence during learning. At the time of classification, they involve only dot products and hence are extremely fast. In terms of recognition accuracy, one variant of the LSM known as the Averaged Learning Subspace Method (ALSM) appears to work best (Oja, 1983, Prakash and Murty, 1995b).

From the point of view of computational power, two properties are associated with all the subspace methods: (i) the classification of a pattern x is based solely on its direction and does not depend on the magnitude of x , and (ii) the decision surfaces are quadratic. Independence of magnitude is desirable in some applications (Oja, 1983, Prakash and Murty, 1995b). However, the second property may be a limitation in some applications. Piecewise linear approximations have been used in the past to overcome similar problems in PR (e.g., Sklansky and Michelotti, 1980, Kovalovsky 1980, Choi and Choi, 1994).

In this paper, we propose the use of a piecewise linear approximation in the subspace methods to overcome the limitation of quadratic decision surfaces. Here, each class is represented by a set of subspaces (clusters). By keeping the number of basis vectors the same among all these clusters, a piecewise linear decision surface can be obtained. We extend the SPRM and the ALSM to obtain the Extended SPRM (ESPRM) and the Extended ALSM (EALSM), respectively. We investigate the effectiveness of these methods here. The organization of the paper is as follows. In Section 2, we briefly present the SPRM. In Sections 3 and 4 we present the ESPRM and the EALSM, respectively. The experimental results are presented in Section 5. Finally, in Section 6, we summarize and conclude.

2. The subspace pattern recognition method

The subspace methods of classification are decision-theoretic pattern recognition methods where the primary model for a class is a linear subspace of the Euclidean pattern space (Watanabe and Pakvasa, 1973). Any set of m linearly independent vectors $u_1, \dots, u_m$ in $\mathbb{R}^n$ (with $m < n$) spans a subspace, L :

$$L = L(u_1, \dots, u_m) \\ = \left\{ x \mid x = \sum_{i=1}^m \alpha_i u_i \text{ for some scalars } \alpha_1, \dots, \alpha_m \right\}.$$

The basic operation on a subspace is a projection of a vector, and is given by $\hat{x} = Px$, where the projection matrix, P , of L is given by

$$P = A(A^T A)^{-1} A^T, \quad A = [u_1, \dots, u_m].$$

$\tilde{x} = x - \hat{x}$ is the corresponding orthogonal residual. The squared orthogonal distance of x from L is defined by

$$d(x, L) = \|\tilde{x}\|^2 / \|x\|^2.$$

This orthogonal distance forms the classification criterion, and x is classified to the class which gives the shortest d .

Given K pattern classes $\omega^1, \dots, \omega^K$, each represented by its subspace $L^i = L(u_1^i, \dots, u_{m^i}^i)$, the classification rule is:

if $d(x, L^i) < d(x, L^j) \quad \forall j \neq i$,
then classify x into ω^i .

Alternatively, the classification rule could also be given in the following form:

if $\delta(x, L^i) > \delta(x, L^j) \quad \forall j \neq i$,
then classify x into ω^i , (1)

where $\delta(x, L^j)$, the squared orthogonal projection distance, is defined as

$$\delta(x, L^j) = \begin{cases} x^T P^j x & \text{when } u_1^j, \dots, u_{m^j}^j \text{ are non-orthonormal,} \\ \sum_{i=1}^{m^j} (x^T u_i^j)^2 & \text{when } u_1^j, \dots, u_{m^j}^j \text{ are orthonormal.} \end{cases} \quad (2)$$

Given any data set, the SPRM classifier accuracy is obtained as follows: we take each sample from this data set and use Eqs. (1) and (2) to obtain the class to which the SPRM assigns it. If this is the same as the given classification label associated with the data sample, it is said to be correctly classified. The SPRM classifier accuracy is defined as the percentage of data samples correctly classified as above.

From Eqs. (1) and (2) we see that the decision surfaces obtained by the SPRM are quadratic, and the classification is independent of the magnitude. We also see that the computation involved in the classification is simplified considerably when the basis vectors chosen are orthonormal. In the CLAFIC algorithm (Watanabe and Pakvasa, 1973), we can choose principal components (PCs), i.e., those eigenvectors of the class covariance/correlation matrix that correspond to a few largest eigenvalues, as basis vectors. This number could either be simply predetermined or may correspond to capturing a given percentage (a threshold) of variance in the data.

3. Extended subspace pattern recognition method

In this method, each class is represented using one or more clusters. Let $C_1^i, \dots, C_{n^i}^i$ represent the n^i clusters of class ω^i . Let $L^{i,j} = L(u_1^{i,j}, \dots, u_{m^{i,j}}^{i,j})$ and $P^{i,j}$ be the subspace and the projection matrix, respectively corresponding to the j th cluster of the i th class. The classification rule given by Eqs. (1) and (2) above is now modified as follows:

$$\text{if } \delta(x, L^{i,j}) > \delta(x, L^{k,l}) \quad \forall (i, j) \neq (k, l),$$

$$\text{then classify } x \text{ into } (\omega^i, C_j^i), \quad (3)$$

where $\delta(x, L^{i,j})$, the squared orthogonal projection distance on the j th cluster of the i th class, is defined as

$$\delta(x, L^{i,j}) = \begin{cases} x^T P^{i,j} x & \text{when } u_1^{i,j}, \dots, u_{m^{i,j}}^{i,j} \text{ are non-orthonormal,} \\ \sum_{k=1}^{m^{i,j}} (x^T u_k^{i,j})^2 & \text{when } u_1^{i,j}, \dots, u_{m^{i,j}}^{i,j} \text{ are orthonormal.} \end{cases} \quad (4)$$

The classification is independent of magnitude as before. The decision surfaces in earlier methods like

the SPRM and the ALSM are usually quadratic. In the special case of equal number of basis vectors, the decision surfaces are linear. By having many clusters in each class, and the same number of basis vectors in all clusters, these extended methods obtain a piecewise linear decision boundary. Thus, any decision boundary can be approximated by choosing an appropriate number of clusters in each class.

In the SPRM, determining the subspace corresponding to any class can simply be obtained using the CLAFIC algorithm based on all the training patterns belonging to that class. However, in the extended methods, as there are more than one cluster in each class, these patterns have to be split and assigned to different clusters on some basis. We have investigated the following two modes of assignment in our experiments: (i) split the available patterns equally among the clusters in a random manner, and (ii) use a dynamic clustering algorithm (Didey and Simon, 1980) to determine the initial clusters.

The idea behind the dynamic clustering algorithm for subspaces is to maximize the total variance captured by all the subspaces, thus resulting in a more faithful representation of the patterns by a given number of subspaces/clusters. Thus, it is likely to improve the classification accuracy. The algorithm is as follows. It starts with a random assignment as in mode (i). Based on this assignment, the initial subspaces are formed (we use CLAFIC). Now, all the patterns are assigned to these clusters again. Based on this new assignment, the next set of clusters is determined as before. This iterative process is carried out either until a stable configuration is reached or for a given number of iterations. Using covariance matrices, the dynamic clustering algorithm always reaches a stable configuration (Didey and Simon, 1980). We use correlation matrices as the ALSM requires them instead of the covariance matrices. However, in all our experiments (to be described in Section 5), stable configurations could be reached.

4. Extended averaged learning subspace method

In the ALSM, sample estimates of *conditional correlation matrices* are used, where the conditioning events are the two types of misclassification: either a sample vector of class ω^i is misclassified into another

class, say ω^j , or a sample vector of another class, say ω^k , is misclassified into class ω^i . Denote the unnormalized conditional correlation matrix estimate by

$$S^{(i,j)} = \sum_x \{xx^T \mid x \in \omega^i, x \mapsto \omega^j\},$$

with $\mapsto$ denoting “gets classified into”. At each step of the iteration, there exist *current subspaces* for all classes, and based on them all data vectors x can be classified and all matrices $S^{(i,j)}$, $i, j = 1, \dots, K$, can be computed. At the beginning of the iterative process, the initial subspaces are usually chosen to correspond to the SPRM.

In case of the EALSM, we have more than one cluster in each class. In case of misclassifications, the EALSM requires a new notion of a correct cluster. When Eqs. (3) and (4) are used, we can find out the wrong class and the wrong cluster to which a sample is assigned. The sample has only a correct class label. The correct cluster is defined simply as that cluster with maximum orthogonal projection as given by Eq. (4) among all the clusters belonging to the correct class. Only the correct cluster and the wrong cluster are *rotated* here. Based on these notions, we extend the class conditional correlation matrices given above as follows:

$$S^{(i,j,k,l)} = \sum_x \{xx^T \mid x \in (\omega^i, C_j^i), x \mapsto (\omega^k, C_l^k)\}.$$

At each step of the iteration, there exist *current subspaces* for all clusters of all classes, and based on them all data vectors x can be classified and all matrices $S^{(i,j,k,l)}$, $i, k = 1, \dots, K$, $j = 1, \dots, n^i$, $l = 1, \dots, n^k$, can be computed. These updated conditional correlation matrices are then used to obtain new estimates of the correlation matrices from which the new subspaces will be formed. This process can be repeated for a given number of iterations. At the beginning of the iterative process, the initial subspaces are chosen to correspond to the ESPRM. We now give the detailed algorithm below.

Algorithm EALSM for constructing the classification subspaces

begin

compute (unnormalized) class correlation matrix estimates

$$\hat{M}^{i,j}, i = 1, \dots, K, j = 1, \dots, n^i;$$

form initial subspaces

$$L^{i,j}, i = 1, \dots, K, j = 1, \dots, n^i$$

with appropriate number of dimensions using CLAFIC

repeat until classification is stable or given number of steps;

for each data vector x **do**

classify x in current subspaces according to Eqs. (3) and (4);

if x is classified incorrectly **then**

let the correct class be i and the wrong class be k ;

let the correct cluster be j and the wrong cluster be l ;

$$\text{update: } S^{(i,j,k,l)} = S^{(i,j,k,l)} + xx^T;$$

endif;

endfor;

for $i = 1$ to K **do**

for $j = 1$ to n^i **do**

$$\hat{M}^{(i,j)} = \hat{M}^{(i,j)} + \sum_{k \neq i, l} \alpha S^{(i,j,k,l)} - \sum_{k \neq i, l} \beta S^{(k,l,i,j)};$$

form initial subspaces

$$L^{i,j}, i = 1, \dots, K, j = 1, \dots, n^i$$

with appropriate number of dimensions using CLAFIC

endfor;

endrepeat;

end

Both α and β are positive constants.

5. Experiments and results

5.1. Data sets used

We have chosen three data sets: the optical character data, the Sonar data and the vowel data.

Character data. This data set consists of handwritten digits, and was used earlier (Alireza and Hong, 1990; Prakash and Murty, 1995b). A wide intraclass variation exists in this data set. Each digit is a binary image of size 32×24 pixels. There are 13326 training and 3333 test samples. Prakash and Murty (1995b) divided the original 13326 training samples into two near-halves: a training set consisting of 6670 samples, and a validation set consisting of 6656 sam-

ples. Keeping the computational resources in mind, especially from the point of view of the ALSM, they reduced the dimensionality as follows: they formed non-overlapping windows of size 2×2 over the entire image, and replaced each window by one feature whose value corresponds to the number of bits on in that window. Thus the value of every feature varies from 0 to 4, and there are a total of 192 features. We have used this modified data here.

Sonar data. This problem addresses the undersea target identification. A set of 208 sonar returns (one class of 111 cylinder returns, and another class of 97 rock returns) were split into a training and a test data set of size 104 each. The input is prepared as follows. A set of sampling apertures is superimposed over the 2-D display of the short-term Fourier transform spectrogram of the sonar return. A spectral envelope is obtained by integrating over each aperture. Sixty sample points were obtained for each envelope. These samples were normalized to take values between 0.0 and 1.0, and these form the input features. For more details, the reader is referred to (Gorman and Sejnowski, 1988).

Vowel data. This problem addresses the identification of 11 steady state vowels of English spoken by fifteen speakers for a speaker normalization study. The speech signals are low-pass filtered at 4.7 kHz and then digitised to 12 bits with a 10 kHz sampling rate. Twelfth-order linear predictive analysis was carried out on six 512-sample Hamming windowed segments from the steady part of the vowel. The reflection coefficients were used to calculate 10 log area parameters, giving a 10-dimensional input space. Extensive results on this data are reported by Fritzke (1993). Both the vowel and the sonar data are electronically available from the Carnegie-Mellon University connectionist benchmark collection (see Fahlman, 1993).

The details of all data sets are summarized in Table 1.

5.2. Results and discussion

We simulated both the ESPRM and the EALSM on all the three data sets. During each study, the number of clusters in every class is kept the same, and is varied from 1 to 3. Note that 1 cluster cases correspond

Table 1
Details of the data sets used

Data set	Number of classes	Dimensionality	Number of patterns		
			Trg	Val	Test
Sonar	2	60	200	–	242
Vowel	11	10	528	–	462
Char	10	192	6670	6656	3333

Table 2
Comparison of results obtained by different subspace methods

Data set	Classification method	Accuracy (%)		
		Trg	Val	Test
Char	SPRM	91.2	88.7	88.9
	ALSM	99.1	90.5	91.1
	ESPRM	95.8	91.3	91.4
	EALSM	100.0	91.9	92.0
Sonar	SPRM	86.5	–	83.6
	ALSM	100.0	–	85.6
	ESPRM	92.3	–	87.5
	EALSM	100.0	–	93.3
Vowel	SPRM	64.0	–	43.3
	ALSM	76.1	–	37.2
	ESPRM	96.6	–	47.6
	EALSM	100.0	–	45.9

to the SPRM and the ALSM. The number of PCs in each cluster is the same for all classes, and these experiments are conducted for six different values of it. For clusters 2 and 3, we started with the initial subspaces obtained by both random assignment of patterns, and assignment based on the dynamic clustering algorithm. The number of iterations in the EALSM is 20, 50 and 50 for the character, the sonar and the vowel data sets, respectively. In case of the dynamic clustering algorithm, the corresponding figures are 10, 20 and 20, respectively. Earlier experience with the ALSM (Oja, 1983) indicates that the suitable range of values for α and β is between 0.5 and 2.0. Based on this, we have chosen the values of $\alpha (= \beta)$ as 0.5 in all cases.

The best results for all the above experiments are summarized in Table 2. The detailed results of all cases are not presented in the form of tables as their interpretation would be difficult. Hence, we have presented them in graphical form. Fig. 1 shows how the classification accuracy varies as the number of PCs in each cluster is varied for both the SPRM, ESPRM, ALSM and the EALSM in case of the character data. Similarly, Figs. 2 and 3 present results for the sonar and the vowel data, respectively. Based on all these results we can observe the following.

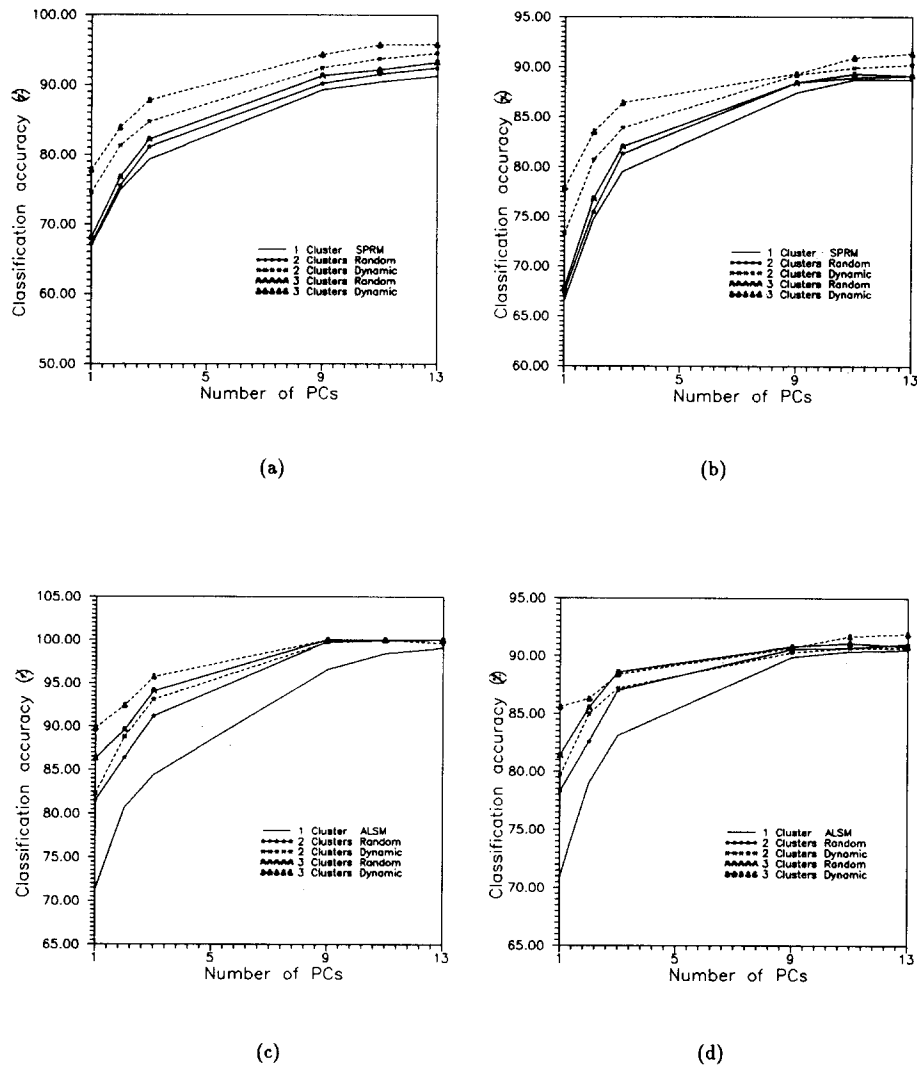


Fig. 1. Number of PCs in each cluster versus classification accuracy for the character data: (a) ESPRM on the training data, (b) ESPRM on the validation data, (c) EALSM on the training data, (d) EALSM on the validation data.

Classification accuracy. From Table 2, we observe that both the extended methods could improve upon their counterparts on all the data sets not only on the training data but also on the test data. It is interesting to note that both the ESPRM and the EALSM gave better results than the ALSM in all cases of testing, including the character and the sonar data where the ALSM gave 100% accuracy on the training data. These results indicate that the representation of a class by more than one subspace is indeed helpful in obtaining better decision surfaces compared to the earlier subspace

methods. Also, the EALSM gave 100% accuracy on all the training data sets, while the ALSM could give only 76.1% on the vowel data. Thus, EALSM appears to have the potential to approximate any decision surface.

Further, from Figs. 1, 2 and 3, we see that the extended methods exhibit an improvement over their counterparts for *all* values of PCs considered except in a very few cases (note that the solid line without the centered symbols is below the other curves in all 12 graphs).

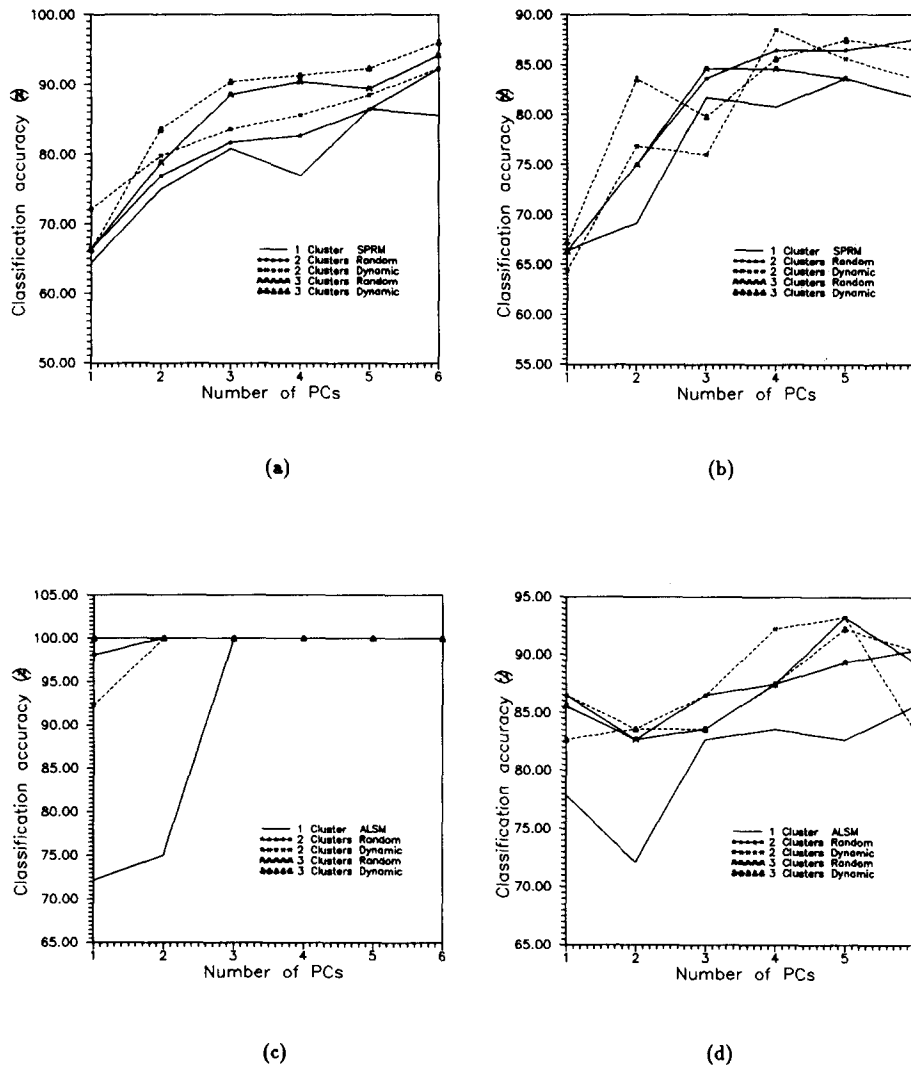


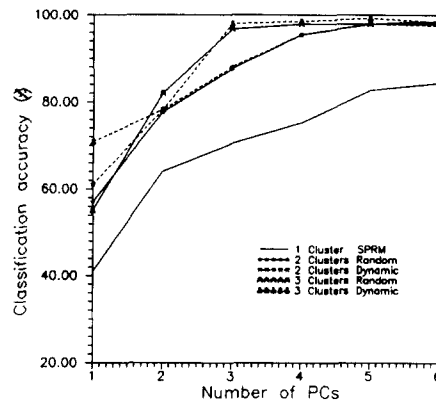
Fig. 2. Number of PCs in each cluster versus classification accuracy for the sonar data: (a) ESPRM on the training data, (b) ESPRM on the test data, (c) EALSM on the training data, (d) EALSM on the test data.

Effect of dynamic clustering algorithm.

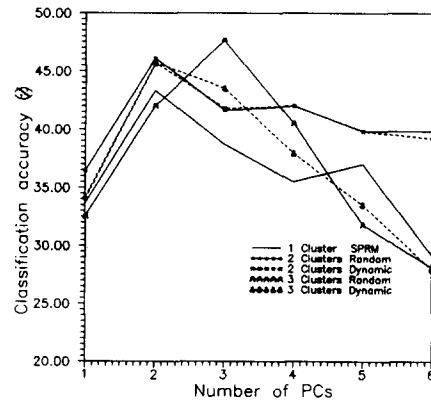
On the ESPRM (Figs. 1(a), 1(b), 2(a), 2(b), 3(a) and 3(b)): The algorithm improved the classification accuracy appreciably for all training data sets. For the test data, such a trend could be seen only for the character data.

On the EALSM (Figs. 1(c), 1(d), 2(c), 2(d), 3(c) and 3(d)): There is no clear trend on either the training or the test data sets.

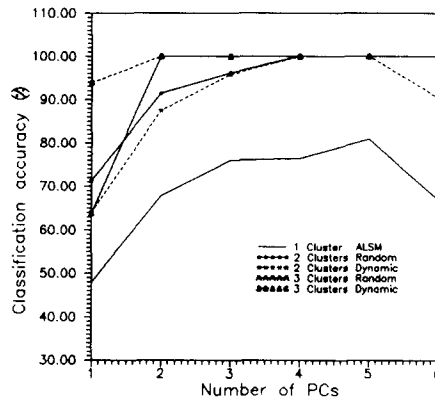
Selection of design parameters. The number of clusters in each class and the number of PCs in each cluster are the design parameters. The ranges of values chosen appear to be sufficient as either the classification accuracy reached an asymptotic value like in the character data or started decreasing for the other data sets. The combined effect of these parameters is erratic on both the sonar and the vowel data. These results indicate that more methodical techniques to choose these parameters are required.



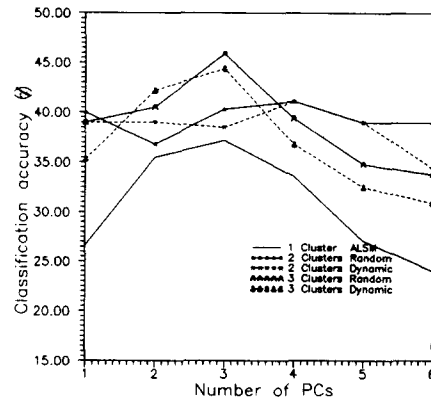
(a)



(b)



(c)



(d)

Fig. 3. Number of PCs in each cluster versus classification accuracy for the vowel data: (a) ESPRM on the training data, (b) ESPRM on the test data, (c) EALSM on the training data, (d) EALSM on the test data.

6. Summary and conclusion

In this paper, we have proposed the use of more than one subspace (cluster) in the representation of the classes in the subspace methods of pattern recognition. By keeping the number of PCs in all clusters the same, this model allows for a piecewise linear approximation of the decision surfaces, and thus helps in overcoming a limitation of having only quadratic decision surfaces. We have used this model to extend both the SPRM and the ALSM to obtain the ESPRM

and the EALSM, respectively. The proposed methods are suitable for those problems where the classification is independent of magnitude and the decision surfaces are highly nonlinear. They include problems that use a binary representation like the character data, and speech and image processing problems that use histograms and spectral features as input. We have conducted experiments on three data sets. The results indicate that the proposed methods can considerably improve the classification accuracy on the test sets. In fact, even ESPRM gave better results than the ALSM.

Also, the EALSM yielded 100% accuracy on all training data sets. These results indicate that the proposed extended methods have the potential to approximate any decision surface. Initializing the clusters with a dynamic clustering algorithm seems to be helpful in some cases. However, choosing appropriate values for the number of clusters in each class and the number of PCs in each cluster appears to be difficult. Hence, more methodical techniques to obtain suitable values for them are desirable, and such techniques may lead to a further improvement in classification accuracy.

Acknowledgements

The authors thank Prof. M.D. Srinath for providing the character data used in this paper. The authors thank the anonymous referees whose comments have helped in improving the quality of the paper.

References

- Alireza, K. and Y.H. Hong (1990). Rotation invariant image recognition using features selected via a systematic method. *Pattern Recognition* 23 (10), 1089–1101.
- Choi, C.H. and J.Y. Choi (1994). Constructive neural networks with piecewise interpolation capabilities for function approximation. *IEEE Trans. Neural Networks* 5, 936–943.
- Diday, E. and J.C. Simon (1980). Clustering analysis. In: K.S. Fu, Ed., *Digital Pattern Recognition*. Springer, Berlin, 47–94.
- Fahlman, S.E. (1993). *CMU Benchmark Collection for Neural Net Learning Algorithms*. Carnegie-Mellon University, School of Computer Science, Pittsburg, PA.
- Fritzke, B. (1993). Growing cell structures – A self-organizing network for unsupervised and supervised learning. Technical Report TR-93-026, Internat. Computer Science Institute, Berkeley, CA.
- Gorman, R.P. and J. Sejnowski (1988). Learned classification of sonar targets using a massively parallel network. *IEEE Trans. Acoust. Speech Signal Process.* 36 (7), 1135–1140.
- Kohonen, T., G. Nemeth, K. Bry, M. Jalanko and K. Makisara (1979). Spectral classification of phonemes by learning subspace methods. *Proc. IEEE Internat. Conf. Acoust. Speech Signal Process.*, Washington, DC, 97–100.
- Kovalevsky, V.A. (1980). *Image Pattern Recognition*. Springer, New York.
- Oja, E. (1983). *Subspace Methods of Pattern Recognition*. Research Studies Press, Letchworth, UK.
- Oja, E. and M. Kuusela (1983). The ALSM algorithm – an improved subspace method of classification. *Pattern Recognition* 16 (4), 421–427.
- Oja, E. and J. Parkkinen (1986). Texture subspaces. In: P.A. Devijver and J. Kittler, Eds., *Pattern Recognition – Theory and Applications*. Springer, Berlin, 21–33.
- Prakash, M. and M.N. Murty (1995a). A genetic approach for selection of (near-) optimal subsets of principal components for discrimination. *Pattern Recognition Lett.* 16, 781–787.
- Prakash, M. and M.N. Murty (1995b). Hebbian learning subspace method: A new approach. *Pattern Recognition*, communicated.
- Sklansky, J. and L. Michelotti (1980). Locally trained piecewise linear classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* 2 (2), 101–110.
- Watanabe, S. (1969). *Knowing and Guessing*. Wiley, New York.
- Watanabe, S. and N. Pakvasa (1973). Subspace method in pattern recognition. *Proc. Internat. Joint Conf. on Pattern Recognition*, 25–32.