*Reprinted from*

Pattern Recognition Letters

Pattern Recognition Letters 15 (1994) 1119–1125

# Floating search methods in feature selection

P. Pudil *, J. Novovičová [1], J. Kittler

*Department of Electronic and Electrical Engineering, University of Surrey, Guildford, Surrey GU2 5XH, United Kingdom*

Received 19 June 1993

NORTH-HOLLAND AMSTERDAM

ELSEVIER

November 1994

Pattern Recognition Letters 15 (1994) 1119–1125

Pattern Recognition Letters

# Floating search methods in feature selection

P. Pudil *, J. Novovičová [1], J. Kittler

*Department of Electronic and Electrical Engineering, University of Surrey, Guildford, Surrey GU2 5XH, United Kingdom*

Received 19 June 1993

**Abstract**

Sequential search methods characterized by a dynamically changing number of features included or eliminated at each step, henceforth "floating" methods, are presented. They are shown to give very good results and to be computationally more effective than the branch and bound method.

*Keywords*: Pattern recognition; Feature selection; Feature ordering; Search methods

## 1. Introduction

The main goal of feature selection is to select a subset of $d$ features from the given set of $D$ measurements, $d<D$, without significantly degrading the performance of the recognition system. Assuming that a suitable criterion function has been chosen to evaluate the effectiveness of feature subsets, feature selection is reduced to a search problem that detects an optimal feature subset based on the selected measure. Therefore, some computationally feasible procedures designed to avoid the exhaustive search are essential even though the feature set obtained may be suboptimal.

For the above reason, the question of the trade-off between the optimality and efficiency of algorithms for feature selection is recognized, and the mainstream of research on feature selection has thus been directed toward suboptimal search methods.

With the feature set search algorithms discussed in this paper the best feature set is constructed by adding to and/or removing from the current feature set, a small number of measurements at a time until the required feature set, $X_d$, of cardinality $d$ is obtained. More specifically, to form the best set of $d$ features, the starting point of the search can be either an empty set, $X_0$, which is then successively built up or the starting point can be the complete set of measurements, $Y$, in which superfluous measurements are successively eliminated. The former approach is referred to as the "bottom up" search while the latter is known as the "top down" method.

A feature selection technique using the divergence distance as the criterion function and the *sequential backward selection* (SBS) method as the search algorithm was introduced already by Marill and Green (1963) and its "bottom up" counterpart known as *sequential forward selection* (SFS) by Whitney (1971). Both these methods are generally suboptimal and suffer from the so-called "nesting effect". It means that in the case of the "top down" search the discarded features cannot be re-selected while in the case of the "bottom up" search the features once se-

* Corresponding author. Email: ees2pp@ee.surrey.ac.uk.

[1] Permanently with the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, 182 08 Prague 8, Czech Republic.

Elsevier Science B.V.
*SSDI* 0167-8655(94)00050-D

lected cannot be later discarded. The result is that the methods are only suboptimal.

An attempt to prevent the nesting of feature subsets was first put forward by Michael and Lin (1973) in the context of Whitney's sequential forward selection. The idea was later refined and developed into the Plus-*l*-Minus-*r* (*l*-*r*) search method (also suboptimal) (Stearns, 1976). The main drawback of this method is that there is no theoretical way of predicting the values of *l* and *r* to achieve the best feature set. The search in this direction was concluded by introducing the generalization of SBS, SFS, and (*l*-*r*) algorithms proposed by Kittler (1978).

A computationally very appealing method was proposed by Backer and Schipper (1977). The method involves only the computation of individual and pairwise merits of features. It employs a search in the sequential forward selection manner and it is known as the *Max-Min* algorithm. However, the results achieved with this method are invariably rather unsatisfactory. A comparative study of various feature selection algorithms made in Kittler (1978) indicates that the Max-Min method gives the poorest results. The results confirm that it is not possible to select a set of features in a high-dimensional space based on two-dimensional information measures without a substantial information loss (Cover and Van Compenhount, 1977). In addition to this, however there are other detrimental factors which are inherent to the Max-Min method itself, as shown by Pudil et al. (1993a).

A real breakthrough in optimal set search came in 1977 with the introduction of the *branch and bound* algorithm which was proposed by Narendra and Fukunaga (1977). The optimality of the results in this method, however, is constrained by the fact that monotonic parametric distance measures (e.g. Bhattacharyya distance, divergence) must be used as the criterion function, i.e., the monotonicity condition must be satisfied. The branch and bound algorithm often makes practicable problems for which the exhaustive search would be totally out of the question. However, even the branch and bound algorithm becomes impracticable for feature selection problems involving more than 30 measurements.

Since the introduction of the branch and bound procedure, the major work on feature selection has been directed toward graph search procedures (Ichino and Sklansky, 1984). The usefulness of this approach is, however, still dependent on the computation speed and computer memory. Further work includes the use of genetic algorithms for feature selection (Siedlecki and Sklansky, 1989) or the possibility of applying simulated annealing technique (Siedlecki and Sklansky, 1988). However, the optimality of the selected feature set from either the genetic algorithm or simulated annealing cannot be guaranteed.

To conclude, despite some progress, the available optimisation search techniques are not yet completely satisfactory. They are either computationally feasible but yield feature subsets which are far from optimal, or they yield optimal or almost optimal feature subsets but cannot cope with the inherent computational complexity of feature selection problems of realistic size.

Therefore, the floating search methods of feature selection presented in this paper should be considered as an alternative intended to overcome the above problems. Though neither these methods can guarantee always to provide the best subset of features, their performance has been found to be very good compared with other search methods and, furthermore, they are computationally much more efficient than the branch and bound method.

The next section of the paper is devoted to a formal description of the floating search methods. Finally, the experimental comparison of the performance of floating search methods with that of the currently used search methods is presented in Section 3.

## 2. Sequential forward floating and sequential backward floating selection methods

A simple way to avoid nesting of feature sets is to employ either the (*l*, *r*) or generalized (*l*, *r*) algorithm which involves successive augmentation and depletion processes. Consequently, the resulting dimensionality in respective stages of both algorithms is fixed depending on the prespecified values of *l* and *r*. Unfortunately, there is no theoretical way of predicting the values of *l* and *r* to achieve the best feature set. Alternatively, instead of fixing these values, we can let these values "float", i.e., to keep them flexibly changing so as to approximate the optimal solution as much as possible. Consequently, the resulting dimensionality in respective stages of the algorithm is not changing monotonously but is actually "floating" up and down.

Because of this "floating" characteristic, the two methods designed and implemented in the PREDITAS software package (Pudil et al., 1991) have been denoted *floating search methods*. Although both these methods switch between including and excluding features, they are based on two different algorithms according to the dominant direction of the search. The search in the forward direction is referred to as the *sequential forward floating selection* (SFFS), while in the opposite direction it will be called the *sequential backward floating selection.*

### 2.1. Preliminaries

Before describing the corresponding algorithms formally, the following definitions have to be introduced.

Let $X_k=\{x_i\colon 1\leqslant i\leqslant k,\ x_i\in Y\}$ be the set of $k$ features from the set $Y=\{y_i\colon 1\leqslant i\leqslant D\}$ of $D$ available features. The value $J(y_i)$ of the feature selection criterion function if only the $i$th feature $y_i$, $i=1,2,\ldots,D$, is used will be called the *individual significance* $S_0(y_i)$ of the feature.

The *significance* $S_{k-1}(x_j)$ *of the feature* $x_j$, $j=1,2,\ldots,k$, *in the set* $X_k$ is defined by

$$S_{k-1}(x_j)=J(X_k)-J(X_k-x_j)\ . \tag{1}$$

The *significance* $S_{k+1}(f_j)$ *of the feature* $f_j$ from the set $Y-X_k$

$$Y-X_k=\{f_i{:}i=1,2,\ldots,D-k, f_i\in Y,\ f_i\neq x_l \text{ for all } x_l\in X_k\}$$

*with respect to the set* $X_k$ is defined by

$$S_{k+1}(f_j)=J(X_k+f_j)-J(X_k)\ . \tag{2}$$

*Remark.* For $k=1$ the term feature significance in the set coincides with the term of individual significance.

We shall say that the feature $x_j$ from the set $X_k$ is

(a) the *most significant* (best) feature *in the set* $X_k$ if

$$S_{k-1}(x_j)=\max_{1\leqslant i\leqslant k} S_{k-1}(x_i)$$

$$\Rightarrow J(X_k-x_j)=\min_{1\leqslant i\leqslant k} J(X_k-x_i)\ , \tag{3}$$

(b) the *least significant* (worst) feature *in the set* $X_k$ if

$$S_{k-1}(x_j)=\min_{1\leqslant i\leqslant k} S_{k-1}(x_i)$$

$$\Rightarrow J(X_k-x_j)=\max_{1\leqslant i\leqslant k} J(X_k-x_i)\ . \tag{4}$$

We shall say that the feature $f_j$ from the set $Y-X_k$ is

(a) the *most significant* (best) feature *with respect to the set* $X_k$ if

$$S_{k+1}(f_j)=\max_{1\leqslant i\leqslant D-k} S_{k+1}(f_i)$$

$$\Rightarrow J(X_k+f_j)=\max_{1\leqslant i\leqslant D-k} J(X_k+f_i)\ , \tag{5}$$

(b) the *least significant* (worst) feature *with respect to the set* $X_k$ if

$$S_{k+1}(f_j)=\min_{1\leqslant i\leqslant D-k} S_{k+1}(f_i)$$

$$\Rightarrow J(X_k+f_j)=\min_{1\leqslant i\leqslant D-k} J(X_k+f_i)\ . \tag{6}$$

### 2.2. The SFFS procedure

The SFFS is basically a bottom up search procedure which includes new features by means of applying the basic SFS procedure starting from the current feature set, followed by a series of successive conditional exclusion of the worst feature in the newly updated set provided a further improvement can be made to the previous sets.

**The Sequential Forward Floating Selection Algorithm**

Suppose $k$ features have already been selected from the complete set of measurements $Y=\{y_j\mid j=1,2,\ldots,D\}$ to form set $X_k$ with the corresponding criterion function $J(X_k)$. In addition, the values of $J(X_i)$ for all preceding subsets of size $i=1,2,\ldots,k-1$, are known and stored.

- *Step* 1 (*Inclusion*). Using the basic SFS method, select feature $x_{k+1}$ from the set of available measurements, $Y-X_k$, to form feature set $X_{k+1}$, i.e., the most significant feature $x_{k+1}$ with respect to the set $X_k$ is added to $X_k$. Therefore

$$X_{k+1} = X_k + x_{k+1} .$$

- *Step* 2 (*Conditional exclusion*). Find the least significant feature in the set $X_{k+1}$. If $x_{k+1}$ is the least significant feature in the set $X_{k+1}$, i.e.

$$J(X_{k+1} - x_{k+1}) \geqslant J(X_{k+1} - x_j), \quad \forall j = 1, 2, ..., k ,$$

then set $k = k+1$ and return to Step 1, but if $x_r$, $1 \leqslant r \leqslant k$, is the least significant feature in the set $X_{k+1}$, i.e.

$$J(X_{k+1} - x_r) > J(X_k) ,$$

then exclude $x_r$ from $X_{k+1}$ to form a new feature set $X'_k$, i.e.

$$X'_k = X_{k+1} - x_r .$$

Note that now $J(X'_k) > J(X_k)$. If $k = 2$, then set $X_k = X'_k$ and $J(X_k) = J(X'_k)$ and return to Step 1, else go to Step 3.

- *Step* 3 (*Continuation of conditional exclusion*). Find the least significant feature $x_s$ in the set $X'_k$. If $J(X'_k - x_s) \leqslant J(X_{k-1})$ then set $X_k = X'_k$, $J(X_k) = J(X'_k)$ and return to Step 1. If $J(X'_k - x_s) > J(X_{k-1})$ then exclude $x_s$ from $X'_k$ to form a newly reduced set $X'_{k-1}$, i.e.

$$X'_{k-1} = X'_k - x_s .$$

Set $k = k-1$. Now if $k = 2$, then set $X_k = X'_k$ and $J(X_k) = J(X'_k)$ and return to Step 1, else repeat Step 3.

The algorithm is initialized by setting $k = 0$ and $X_0 = \emptyset$, and the SFS method is used until a feature set of cardinality 2 is obtained. Then the algorithm continues with Step 1.

### 2.3. The SBFS procedure

The SBFS is a top down search procedure which excludes features by means of applying the basic SBS procedure starting from the current feature set and followed by a series of successive conditional inclusions of the most significant feature from the available features if an improvement can be made to the previous sets.

**The Sequential Backward Floating Selection Algorithm**

Suppose $k$ features have already been removed from the complete set of measurements $\bar{X}_0 = Y$ to form feature set $\bar{X}_k$ with the corresponding criterion function $J(\bar{X}_k)$. Futhermore, the values of all supersets $\bar{X}_i$, $i = 1, 2, ..., k-1$, are known and stored.

- *Step* 1 (*Exclusion*). Use the basic SBS method to remove feature $x_{k+1}$ from the current set $\bar{X}_k$ to form a reduced feature set $\bar{X}_{k+1}$, i.e., the least significant feature $x_{k+1}$ is deleted from the set $\bar{X}_k$.
- *Step* 2 (*Conditional inclusion*). Find among the excluded features the most significant feature with respect to the set $\bar{X}_{k+1}$. If $x_{k+1}$ is the most significant feature with respect to $\bar{X}_{k+1}$, i.e.

$$J(\bar{X}_{k+1} + x_{k+1}) \geqslant J(\bar{X}_{k+1} + x_j), \quad \forall j = 1, 2, ..., k ,$$

then set $k = k+1$ and return to Step 1. If $x_r$, $1 \leqslant r \leqslant k$, is the most significant feature with respect to the set $\bar{X}_{k+1}$, i.e.

$$J(\bar{X}_{k+1} + x_r) > J(\bar{X}_k) ,$$

then include $x_r$ to the set $\bar{X}_{k+1}$ to form a new feature set $\bar{X}'_k$, i.e.

$$\bar{X}'_k = \bar{X}_{k+1} + x_r .$$

Note that now $J(X'_k) > J(X_k)$. If $k = 2$, then set $\bar{X}_k = \bar{X}'_k$ and $J(\bar{X}_k) = J(\bar{X}'_k)$ and return to Step 1, else go to Step 3.

- *Step* 3 (*Continuation of conditional inclusion*). Find among the excluded features the most significant feature $x_s$ with respect to the set $\bar{X}'_k$. If $J(\bar{X}'_k + x_s) \leqslant J(\bar{X}_{k-1})$ then set $\bar{X}'_k = \bar{X}_k$, $J(\bar{X}'_k) = J(\bar{X}_k)$ and return to Step 1. If $J(\bar{X}'_k + x_s) > J(\bar{X}_{k-1})$ then include $x_s$ to the set $\bar{X}'_k$ to form the new enlarged set $\bar{X}'_{k-1}$, i.e.

$$\bar{X}'_{k-1} = \bar{X}'_k + x_s .$$

Set $k = k-1$. Now if $k = 2$, then set $\bar{X}_k = \bar{X}'_k$ and $J(\bar{X}_k) = J(\bar{X}'_k)$ and return to Step 1, else repeat Step 3.

The algorithm is initialized by setting $k = 0$ and $\bar{X}_0 = Y$ and the SBS method is used until a feature set of cardinality $D-2$ is obtained (it means until the 2 least significant features are excluded). Then the algorithm continues with Step 1.

Unlike the $(l, r)$ and generalized $(l, r)$ algorithms in which factors such as the net change in the size of the current feature set, and especially the amount of computational time, are governed by the values of $l$ and $r$, the SFFS and SBFS methods are not restricted by these factors. By means of conditional "floating down and up" both the methods are freely allowed to correct wrong decisions made in the previous steps so as to approximate the optimal solution as much as possible. Obviously, this near optimality is achieved at the expense of computational time, especially in the case of data of greater complexity and dimensionality. However, as we shall see from the results, both the methods are much faster than the branch and bound method.

## 3. Experimental results and discussion

The described search methods have been evaluated by experiments on various types of data. The results reported in (Choakjarernwanit, 1991) confirm that on simple feature selection problems the relative performance of the various methods is similar. The methods differ only in computational efficiency. In order to clearly demonstrate the effectiveness of each method, the selection of a feature set from data showing high statistical dependencies provides a more discriminative test. Consequently the emphasis herein is put on an experiment involving a nondestructive testing problem data used in (Kittler, 1978) with strong interactions among features. This data consists of two normally distributed classes in a 20-dimensional space with means $\mu_i$, $i=1, 2$, and an equal covariance matrix $\Sigma$. Consequently, it is pertinent to use the Mahalanobis distance $J_M$ as a criterion of feature set effectiveness. Unfortunately, the comparison of the effectiveness of all described search methods together is rather difficult. It will, therefore, be done separately in smaller groups. Note that all the experiments were performed on SUN SPARC station 1.

The results which are partially documented in Figs. 1 and 2 where the results marked "optimal" have been obtained using the branch and bound method can be

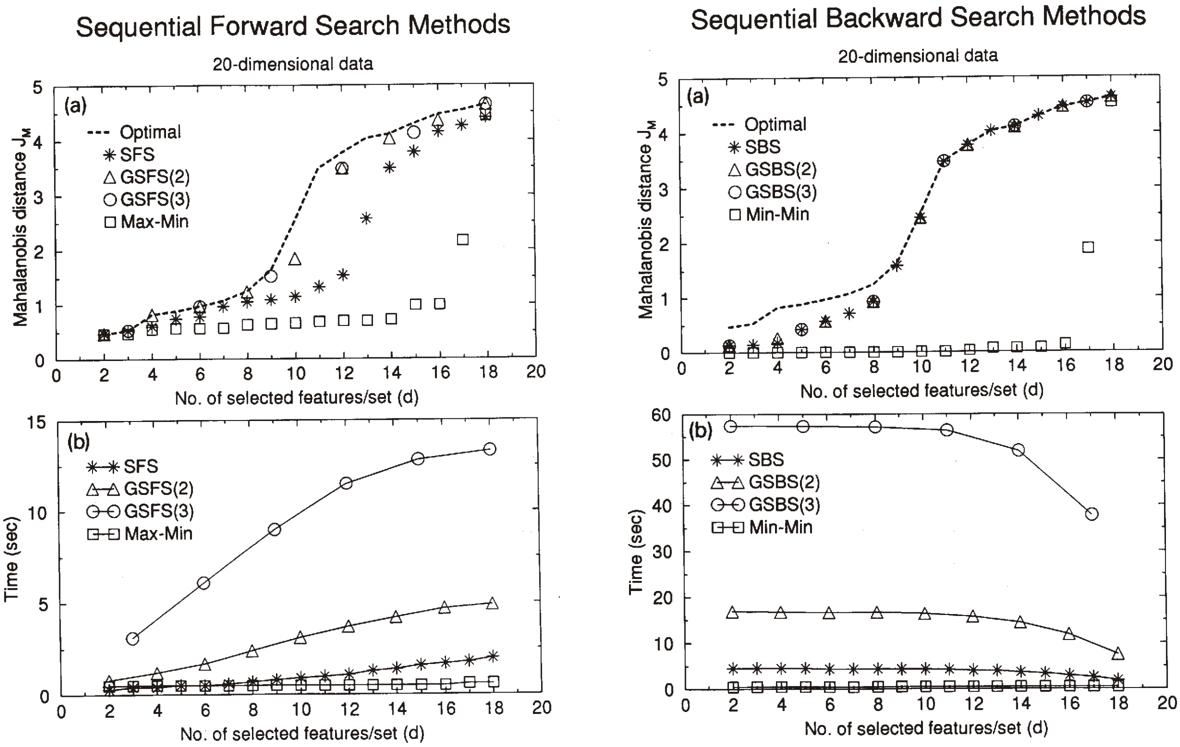

Fig. 1. Results of sequential forward and backward search methods.

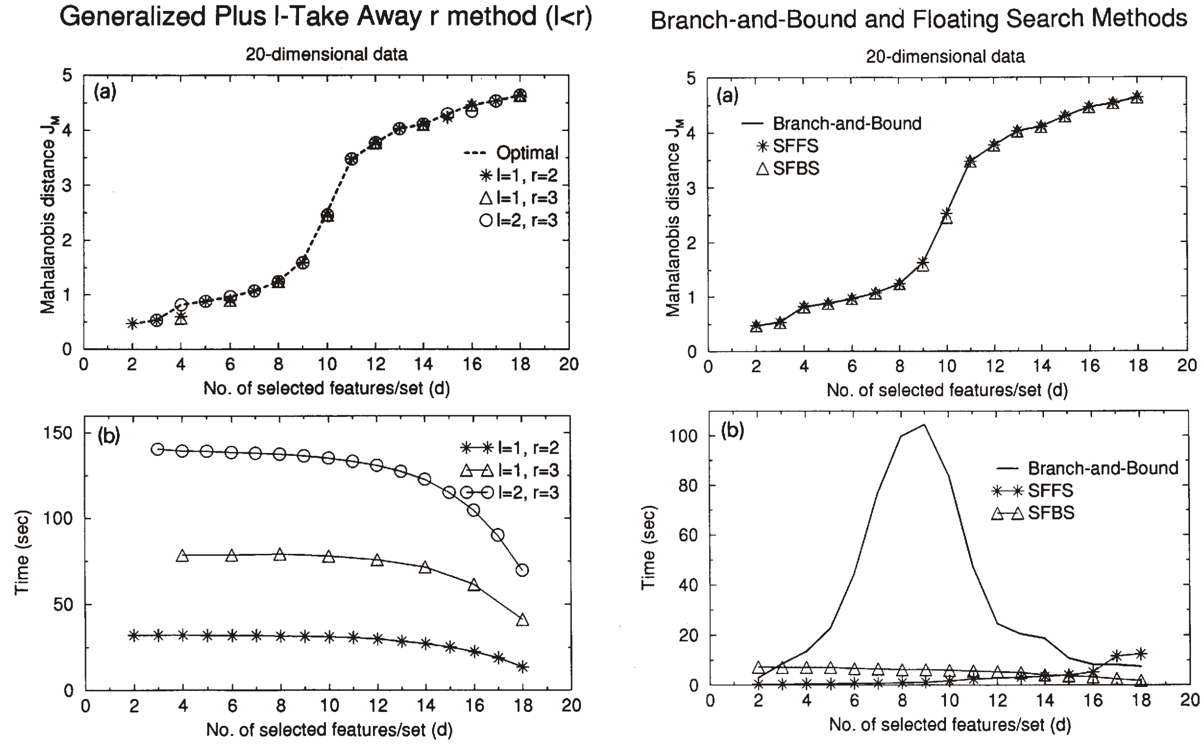

Fig. 2. Results of the generalized (*l*, *r*), branch-and-bound and floating search methods.

summarized as follows (for more detail see (Pudil et al., 1992).

1. Since the Mahalanobis distance satisfies the monotonicity condition, the branch and bound method provides the optimal solution. Though it is much faster than the exhaustive search, its use for feature space of high dimensionality still remains prohibitive. This result is consistent with the findings of Siedlecki and Sklansky (1988).

2. Individual suboptimal sequential forward methods yielded relatively similar results to their backward counterparts, obviously with the exception of the computational time for a given dimensionality which depends in both approaches strongly on the difference between the cardinalities of the feature sets at the initial and final stages of the search process respectively.

3. The "simple" search methods Max-Min, SFS, SBS, (+*l*, −*r*) are relatively fast but fail to provide optimal results, with the Max-Min method giving the worst results. The (+*l*, −*r*) method gives significantly better results than SFS and SBS methods (Pudil et al., 1992).

4. The more sophisticated generalized search methods GSFS(*l*), GSBS(*r*) and G(+*l*, −*r*) yield better results. However, this is at the expense of a much longer computation time, especially with increasing values of parameters *l*, *r*. Moreover, there is no way to determine the right values of *l*, *r* in order to acquire the best feature subset of desired cardinality.

5. Both the SFFS and SBFS methods consistently yield results comparable to the branch and bound method. However, they are computationally much faster. Also the comparison of computation time with other search methods is very favourable for the floating search methods.

## 4. Conclusion

The results achieved so far on various sets of data demonstrate clearly a great potential of the floating search strategies (Pudil et al., 1992, 1993b). They not only provide either the optimal or a close to optimal solution, but also require much less computational time than the branch and bound method and most other currently used suboptimal strategies.

The computational efficiency allows the use of floating search even when the number of original features approaches one-hundred.

Beside of avoiding the nesting of features, one of their distinctive characteristics is that during the backtracking process the values of the criterion function are always compared only with those related to the *same* cardinality of the feature subset. Consequently, a possible decrease of the criterion function when a new feature is added is of no concern. Thus, as opposed to the branch and bound method, the floating search methods are also tolerant to deviations from monotonic behaviour of the feature selection criterion function.

## Acknowledgements

The work has been supported by SERC grant GR/E97549. The authors thank N. Choakjarernwanit for invaluable assistance in programming the methods and in experimentation.

## References

Backer, E. and J.A.D. Schipper (1977). On the max-min approach for feature ordering and selection. In: *The Seminar on Pattern Recognition*, Liège University, Sart-Tilman, Belgium, p. 2.4.1.

Choakjarernwanit, N. (1991). Feature selection in pattern recognition. Technical Report VSSP-TR-1/91, University of Surrey, UK.

Choakjarernwanit, N., P. Pudil and J. Kittler (1991). A detailed study of the max-min in pattern recognition. Technical Report VSSP-TR-5/91, University of Surrey, UK.

Ichino, M. and J. Sklansky (1984). Optimum feature selection by zero-one integer programming. *IEEE Trans. Syst. Man Cybernet.* 14 (5), 737–746.

Kittler, J. (1978). Feature set search algorithms. In: C.H. Chen, Ed., *Pattern Recognition and Signal Processing.* Sijthoff and Noordhoff, Alphen aan den Rijn, Netherlands, 41–60.

Marill, T. and D.M. Green (1963). On the effectiveness of receptors in recognition system. *IEEE Trans. Inform. Theory* 9, 11–17.

Michael, M. and W.C. Lin (1973). Experimental study of information and inter-intra class distance ratios and feature selection and orderings. *IEEE Trans. Syst. Man Cybernet.* 3, 172–181.

Narendra, P.M. and K. Fukunaga (1977). A branch and bound algorithm for feature subset selection. *IEEE Trans. Comput.* 26, 917–922.

Pudil, P., J. Novovičová and S. Bláha (1991). Statistical approach to pattern recognition: Theory and practical solution by means of PREDITAS system. *Kybernetika* 27, Supplement, 1–78.

Pudil, P., J. Novovičová, N. Choakjarernwanit and J. Kittler (1992). A comparative evaluation of floating search methods for feature selection. Technical Report VSSP-TR-5/92, University of Surrey, UK.

Pudil, P., J. Novovičová, N. Choakjarernwanit and J. Kittler (1993a). An analysis of the max-min approach to feature selection. *Pattern Recognition Lett.* 14, 841–847.

Pudil, P., J. Novovičová and J. Kittler (1993b). Computationally efficient quasioptimal algorithms for feature selection. *Proc. Czech Pattern Recognition Workshop* '93, submitted.

Siedlecki, W. and J. Sklansky (1988). On automatic feature selection. *Internat. J. Pattern Recognition and Artificial Intelligence* 2 (2), 197–220.

Siedlecki, W. and J. Sklansky (1989). A note on genetic algorithm for large-scale feature selection. *Pattern Recognition Lett.* 10 (5), 335–347.

Stearns, S.D. (1976). On selecting features for pattern classifiers. *Third Internat. Conf. on Pattern Recognition*, Coronado, CA, 71–75.

Whitney, A.W. (1971). A direct method of nonparametric measurement selection. *IEEE Trans. Comput.* 20, 1100–1103.