# The Behavior Knowledge Space Fusion Method: Analysis of Generalization Error and Strategies for Performance Improvement

Sarunas Raudys[1] and Fabio Roli[2]

[1]Vilnius Gediminas Technical University
Sauletekio 11, Vilnius LT-2100, Lithuania 2006
raudys@ktl.mii.lt
[2]Dept. of Electrical and Electronic Engineering, University of Cagliari
Piazza d'Armi, I-09123 Cagliari, Italy
roli@diee.unica.it

**Abstract.** In the pattern recognition literature, Huang and Suen introduced the "multinomial" rule for fusion of multiple classifiers under the name of Behavior Knowledge Space (BKS) method [1]. This classifier fusion method can provide very good performances if large and representative data sets are available. Otherwise over fitting is likely to occur, and the generalization error quickly increases. In spite of this crucial small sample size problem, analytical models of BKS generalization error are currently not available. In this paper, the generalization error of BKS method is analysed, and a simple analytical model that relates error to sample size is proposed. In addition, a strategy for improving performances by using linear classifiers in "ambiguous" cells of BKS table is described. Preliminary experiments on synthetic and real data sets are reported.

## 1 Introduction

Methods for fusing multiple classifiers can be subdivided according to the types of outputs that can be produced by the individual classifiers: abstract level or single class output, ranked list of classes, and measurement level outputs [2]. Among the methods that work with abstract-level outputs, the Behavior Knowledge Space (BKS) method became very popular [1]. In the BKS method, every possible combination of abstract-level classifiers outputs is regarded as a cell in a look-up table. The BKS table is designed by a training set. Each cell contains the samples of the training set characterized by a particular value of class labels. The training samples in each cell are subdivided per class, and the most representative class label ("majority" class) is selected for each cell. For each unknown test pattern, the classification is performed according to the class label of the BKS cell indexed by the classifiers outputs. From a statistical viewpoint, BKS method tries to estimate high order distribution of classifiers outputs from the frequencies of occurrence in the training set. Further details on BKS method are given in Section 2.1. Differently from other popular fusion methods, BKS does not require any model of classifiers' dependency; in particular, it does not assume conditional independence. On the other hand, some BKS drawbacks are well known. Among the others:

- BKS suffers the small sample size problem. Large and representative data sets are required to estimate high order distribution of classifiers outputs. Otherwise overfitting is likely to occur, and the generalization error quickly increases;
- BKS produces high error for cells characterized by a low probability of the most representative class. (Hereafter, we'll use the adjective "ambiguous" for denoting this type of cells). This is an intrinsic limitation of BKS fusion method.

Some authors proposed techniques for dealing with BKS small sample size problem and limiting error. Reject option is basically used to limit error due to ambiguous cells [1]. Roli et al. showed that increasing the training set size by k-nn noise injection can reduce small sample size problem [4]. Kang and Lee proposed to approximate a high order distribution with a product of low order distributions [7,8]. It is worth noting that, in spite of BKS crucial small sample size problem, analytical models of BKS generalization error are currently not available.

In this paper, the generalization error of BKS method is analysed and a simple analytical model that relates error to sample size in a single cell is proposed (Section 2). In addition, a strategy for improving performances by using linear classifiers in "ambiguous" cells of BKS table is described (Section 3). Preliminary experiments on synthetic and real data sets are reported (Section 3).

## 2 Analysis of BKS generalization error

### 2.1 The BKS fusion method

Let us consider a classification task for $K$ data classes $\omega_1,..,\omega_K$. Each class is assumed to represent a set of specific patterns, each pattern being characterised by a feature vector X. Let us also assume that $L$ different classifiers, $e_j, j = 1,..,L$, have been trained separately to solve the classification task at hand. Let $e_j(X) \in \{1,.., K\}$ indicate the class label assigned to pattern X by classifier $e_j$ . In the BKS method, only class labels are considered (i.e., it is a fusion method for abstract-level classifiers [2]). To simplify the notation, we replace $e_j(X)$ with $e_j$. For each pattern, X, the discrete-valued vector $E=(e_1, e_2....e_L)^T$ of classifiers' outputs can be computed. The number of possible combinations of $L$ classifiers outputs is $m=K^L$. Therefore, the vector $E$ can take one of these $m$ values. In the BKS method proposed by Huang and Suen [1], every possible combination of $L$ class labels is regarded as a cell in a look-up table (BKS table). The BKS table is designed by a training set. Each cell contains the samples of the training set characterized by the particular value of $E$. The training samples in each cell are subdivided per class, and the most representative class label is selected for each cell. For each unknown test pattern, the classification is performed according to the class label of the BKS cell indexed by $E=(e_1, e_2.... E_L)^T$. Obviously, reject option is used for test patterns that fall in empty cells. A threshold on the probability of the most representative class is also used to control the reliability of the decision made in each cell. Basically, the Chow's rule is used to reject patterns of BKS cells with a probability of the most representative class lower than a given threshold ("ambiguous" cells) [5].

For the purposes of this work, let us introduce the following probabilistic view on BKS method. As previously pointed out, each vector $E=(e_1, e_2.... e_L)^T$ can take one of $m$ possible "states" $s_1, s_2, \dots, s_{m-1}, s_m$, $m=K^L$. It is worth noting that, in statistical data analysis, it is supposed that values $s_1,\dots, s_m$ follow the Multinomial distribution. The conditional distribution of the $i$-th class vector $E$ is characterized by $m$ probabilities:

$$P_1^{(i)}, P_2^{(i)}, \dots, P_{m-1}^{(i)}, P_m^{(i)}, \text{ with } \sum_{j=1}^{m} P_j^{(i)} = 1, (i=1, \dots, K). \quad \textbf{(1)}$$

To simplify the notation, we are indicating with the term $P_j^{(i)}$ the conditional probability $p_j(X/\omega_i)$, that is, the conditional distribution of the $i$-th class vector $E$ for patterns falling in the state (BKS cell) $s_j$.

Let $P_i$ be the prior probability of the class $\omega_i$. According to Bayes rule, patterns falling into the state $s_j$ (j=1,…,m) are assigned to the class that maximizes $P_i P_j^{(i)}$ ($i$ =1,...., $K$). This points out that design of BKS rule needs the knowledge of $Kx(m-1)$ probabilities. If such probabilities are exactly known, this is the optimal fusion rule for abstract-level classifiers. In practice, the $Kx(m-1)$ probabilities are unknown and must be estimated from the data. One can use the maximum likelihood estimates (sample frequencies):

$$\hat{P}_j^{(i)} = n_j^{(i)} / N_i \quad \textbf{(2)}$$

where $N_i$ is the number of training vectors belonging to the $i$-th pattern class, and $n_j^{(i)}$ is the number of vectors falling into the state $s_j$. If $\hat{P}_i = N_i / \sum_{i=1}^{L} N_i$ and estimate of prior probability $P_i$ is proportional to $N_i$, we have the sample-based multinomial classifier, that Huang and Suen named BKS method [1]. It is worth remarking that BKS method coincides with the multinomial statistical classifier, that is the optimal statistical decision rule for discrete-valued feature vectors [9]. BKS should be therefore regarded as the application of the multinomial rule to fusion of abstract-level classifiers.

If the values of $L$ and $K$ are large, we have a great number of probabilities to be estimated. To design a good fusion rule, a sufficient number of training vectors should be available to estimate the probabilities in BKS cells. If the size of the data set is not sufficient, the BKS method becomes unreliable. An important issue is therefore to evaluate the size of data set that is necessary for reliable probability estimates. However, to the best of our knowledge, no model was proposed to analyse the generalization error of BKS method as function of sample size. Analysis of BKS generalization error could, first of all, contribute to improve the understanding of this popular fusion rule. In addition, guidelines on the number of samples necessary to limit BKS generalization error could be obtained. In the following section, a simple analytical model that relates BKS error to sample size is proposed for a two-class case, and some implications for BKS table design are discussed.

Finally, it is worth noting that, in small sample cases, small number of classifiers should be preferred. However, if the number of classifiers is small, the cells can become "large" and contain vectors of different classes, that is, ambiguous cells can exist. As pointed out in Section 1, BKS method does not work well for ambiguous cells. In Section 3, we propose a strategy for addressing this limitation of BKS method.

### 2.2 Analysis of generalization error in BKS cells

Consider a two-class problem and $N$ validation patterns belonging to such classes ($N = N_1 + N_2$). Let $N_1$ and $N_2$ be random variables whose distribution depends on class prior probabilities $P_1$ and $P_2$. Prior probabilities are estimated by $\hat{P}_i = N_i/N$. Consider the probability of misclassification for patterns falling into the $j$-th BKS cell ($j$ =1,…, $m$). If the maximum likelihood estimates (equation 2) are used, the local generalization error $P_{err}^{BKS}(j)$ in the $j$-th BKS cell can be written as:

$$P_{err}^{BKS}(j) = Prob\{\hat{P}_1\hat{P}_j^{(1)} < \hat{P}_2\hat{P}_j^{(2)} \mid P_1, P_j^{(1)}, P_j^{(2)}\}\; P_1 P_j^{(1)} + \tag{3}$$

$$Prob\{\hat{P}_1\hat{P}_j^{(1)} > \hat{P}_2\hat{P}_j^{(2)} \mid P_1, P_j^{(1)}, P_j^{(2)}\}\; P_2 P_j^{(2)} + Prob\{\hat{P}_1\hat{P}_j^{(1)} = \hat{P}_2\hat{P}_j^{(2)} \mid P_1, P_j^{(1)}, P_j^{(2)}\}\; (P_1 P_j^{(1)} + P_2 P_j^{(2)})/2$$

Let us remark that we are assuming that an independent validation set is used to estimate probabilities, that is, the $N$ validation patterns do not belong to the set used for training the $L$ classifiers [3].

Random variables $n_j^{(i)}$ can take values 0, 1, 2, 3,…, $N$ (equation. 2). According to definitions given above, sample-based estimates $\hat{P}_i\hat{P}_j^{(i)} = n_j^{(i)}/N$ are multinomial random variables with parameters $q_j^{(1)} = P_1 P_j^{(1)}$, $q_j^{(2)} = P_2 P_j^{(1)}$ , $1 - q_j^{(1)} - q_j^{(2)}$, and $N$ ($j$ =1, 2, … , $m$) . Accordingly, the conditional probabilities in equation 3 can be expressed as follow:

$$Prob\{\,\hat{P}_1\hat{P}_j^{(1)} = r/N,\ \hat{P}_2\hat{P}_j^{(2)} = s/N \mid P_1, P_j^{(1)},\ P_j^{(2)}\,\} = Prob\{\,n_j^{(1)} = r,\ n_j^{(2)} = s \tag{4}$$

$$\mid q_j^{(1)}, q_j^{(2)}\,\} = \frac{N!}{r!s!(N-r-s)!}(q_j^{(1)})^r (q_j^{(2)})^s (1-q_j^{(1)}-q_j^{(2)})^{N-r-s}\,.$$

Therefore, we can write the following equations:

$$Prob\{\,\hat{P}_1\hat{P}_j^{(1)} < \hat{P}_2\hat{P}_j^{(2)} \mid q_j^{(1)}, q_j^{(2)}\,\} = \tag{5a}$$

$$\sum_{r=0=}^{N-1}\sum_{s=r+1}^{N-r}\frac{N!}{r!s!(N-r-s)!}(q_j^{(1)})^r (q_j^{(2)})^s (1-q_j^{(1)}-q_j^{(2)})^{N-r-s}$$

$$Prob\{\,\hat{P}_1\hat{P}_j^{(1)} > \hat{P}_2\hat{P}_j^{(2)} \mid q_j^{(1)}, q_j^{(2)}\,\} = \tag{5b}$$

$$\sum_{s=0=}^{N-1}\sum_{r=s+1}^{N-s}\frac{N!}{r!s!(N-r-s)!}(q_j^{(1)})^r (q_j^{(2)})^s (1-q_j^{(1)}-q_j^{(2)})^{N-r-s}$$

$$Prob\{\,\hat{P}_1\hat{P}_j^{(1)} = \hat{P}_2\hat{P}_j^{(2)} \mid q_j^{(1)}, q_j^{(2)}\,\} = \tag{5c}$$

$$\sum_{r=0}^{N/2}\frac{N!}{r!r!(N-2r)!}(q_j^{(1)})^r (q_j^{(2)})^r (1-q_j^{(1)}-q_j^{(2)})^{N-2r}$$

Substituting equations 5a, 5b, and 5c in equation 3, one obtain the expression that allows evaluating generalization error as function of number of validation patterns, and probabilities $q_j^{(1)} = P_1\,P_j^{(1)}$, $q_j^{(2)} = P_2\,P_j^{(1)}$ . Note that $q_j^{(i)}$ is the probability that patterns of *i-th* class fall in BKS cell *j*.

For the sake of brevity, let us analyze the derived expression of BKS cell generalization error by means of Figure 1. Figure 1 plots the BKS generalization error in a cell as function of the number *N* of samples used for estimating probabilities, that is, for designing the BKS table. The three plots refers to three different values of $q_j^{(1)} = P_1\,P_j^{(1)}$ and $q_j^{(2)} = P_2\,P_j^{(1)}$. Plot number 1: $q_j^{(1)}$ =0.05, $q_j^{(2)}$ =0.05. Plot number 2: $q_j^{(1)}$ =0.05, $q_j^{(2)}$ =0.01. Plot number 3: $q_j^{(1)}$ =0.02, $q_j^{(2)}$ =0.01.

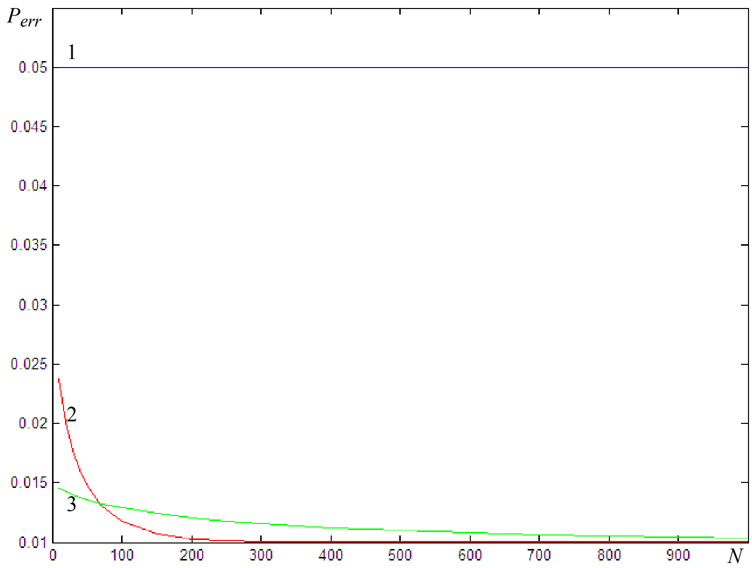


**Fig. 1.** BKS generalization error probability in a cell as function of the number $N$ of samples used for estimating cell conditional probabilities $q_j^{(1)}$ and $q_j^{(2)}$.

For plots "2" and "3", generalization error decreases by increasing the number of samples, and converges to the minimum Bayes error, that is equal to 0.01. Note that $q_j^{(2)}$ =0.01 in the case of plots 2 and 3. BKS assigns all patterns in the cell to the most representative class; therefore, the Bayes error is equal to the probability of the minority class in the cell. The generalization error does not decrease in the case of plot 1 because $q_j^{(1)} = q_j^{(2)}$. It is worth noting that the generalization error decreases quicker for the plot number 2 than for the plot number 3. The reason is that the probability of the most representative class is higher for the case of plot 2 ( $q_j^{(1)}$ =0.05 vs $q_j^{(1)}$ =0.02). On the other hand, the error is higher for small sample sizes in the case of the plot "2".

On the basis of this analysis of BKS generalization error, some preliminary comments can be made:

- Ambiguous cells, characterized by similar class probabilities, negatively affect BKS performances. In particular, our analysis points out that, for such cells, no benefit is obtained by increasing the sample size. Therefore, patterns falling in such cells should be rejected;
- Increasing the sample size is more effective for cells characterized by a class with very high probability, as the error decreases more quickly;
- On the other hand, error is higher for such cells in the small sample case.

With regard to the design of BKS table, our analysis of generalization error suggests that a good table should be made up of cells characterized by a class with very high probability, so that increasing the sample size can give benefits. Design of such a table depends on the selection of classifiers, and it is a topic of further investigations.

## 3 Improvement of BKS accuracy using linear classifiers in ambiguous cells

As pointed out in Section 1, BKS method exhibits an intrinsic limitation for the so called ambiguous cells, that is, for cells characterized by a low probability of the most representative class. As BKS method assigns all the patterns of the cell to the majority class, patterns falling in ambiguous cells are usually rejected in order to limit generalization error [1]. This can cause high reject rates if many ambiguous cells are present. In principle, increasing the number of classifiers could reduce the number of ambiguous cells. Unfortunately, increasing the number $L$ of classifiers makes more difficult the design of BKS table, as the number $m$ of probabilities to be estimated increases exponentially ($m=K^L$). It is worth noting that our analysis of BKS generalization error further pointed out the need for alternative strategies to handle ambiguous cells (Section 2.2), as it showed that small benefits are obtained by increasing the sample size. Therefore, in the following, we propose an alternative strategy that allows exploiting possible increases of sample size, and avoids increasing exponentially the number of probabilities.

First of all, let us indicate with $\{X_j\} \in s_j$ the set of patterns falling in BKS cell $s_j$, j=1,…,m, $m=K^L$. Without loosing of generality, we can assume that such set falls in a region of the original feature space that can be indicated by $\{X_j\} \in \bigcup_{i=1}^{r_j} R_{ji}$. Namely, the set of pattern falling in BKS cell $s_j$ belongs to the union of $R_{ji}$ regions in the original feature space. Now, let us focus on ambiguous cells. For such cells, the majority decision rule of BKS method produces an intrinsic large error. However, it is easy to see that patterns falling in ambiguous cells might be discriminated correctly in the original feature space, supposed that an appropriate discrimination function is used, and enough training patterns are available. Consider, for example, the case of a two-class problem and an ambiguous cell $s_{j^*}$. with 49% of error (i.e., 49% of patterns belongs to the minority class). Assume that patterns of such cell belong to a single region $R_{j^*}$ (i.e., $\{X_{j^*}\} \in R_{j^*}$) in the original feature space, and they are linearly separable within $R_{j^*}$. In such case, a simple linear classifier could discriminate with zero error the patterns of this ambiguous cell, so overcoming the intrinsic limitation of BKS method for such cell.

Therefore, we propose to discriminate patterns falling in ambiguous BKS cells by using additional "local" classifiers that work in the regions of the original feature space associated to such cells. The idea of refining BKS method by local analysis in the original feature space was originally proposed in [10]. In general, classifiers of appropriate complexity should be used. However, due to small sample size problems, we propose the use of linear classifiers. In particular, the use of single layer perceptrons (SLPs) according to the approach proposed in [9]. In order to handle the small sample size problem, data are first moved into the centre of coordinates, and data whitening, normalization of variances are performed. Then, SLP learning can start from weights with zero value, and a variety of classifiers of different complexity can be obtained while training proceeds. Details on this approach can be found in Chapter 5 of [9].

It is easy to see that the use of SLPs, according to the approach proposed in Chapter 5 of [9], can strongly reduce BKS error if patterns falling in ambiguous cells are almost linearly separable in the original feature space. This is more likely to happen for BKS cells whose patterns fall into a single compact region of the original feature space. For ambiguous cells whose patterns fall in a set of disjoint regions, the use of more complex classifiers should be investigated.

Finally, it is worth noting that our analysis of BKS generalization error pointed out that, for ambiguous cells, small benefits are obtained by increasing the sample size. Differently, SLPs can obviously benefit from the increase of sample size.

## 4 Experimental results

Our experiments were aimed to assess the performance improvement achievable by the technique proposed in the previous section. For the sake of brevity, we report only two experiments, with an artificial data set and a real data set of remote sensing images.

### 4.1 Experiments with the artificial data set

The artificial data set was explicitly designed to assess the effectiveness of the proposed technique to increase BKS accuracy for ambiguous cells. Therefore, this data set was designed to favour the presence of ambiguous cells. The data set is shown in Figure 2. It is made up of five “balls”, each ball containing patterns of two non-linearly separable classes. The data set is thirty dimensional, but all the discriminatory information is concentrated in only two dimensions. Figure 2 shows a projection of test data on the two-dimensional informative feature space. For training, we used two thousand 30-dimensional vectors. In order to favour the presence of ambiguous cells, we subdivided training data in five clusters by k-means clustering algorithm (k=5). Then, five multiplayer perceptron (MLP) classifiers, with two hidden nodes, were separately trained on each cluster by the Levenberg-Marquardt algorithm in order to obtain an ensemble of five “specialised” classifiers that were likely to generate ambiguous cells. (It is quite easy to see that classifiers trained on separate data sets are likely to generate ambiguous BKS cells on test data). To avoid over fitting of neural nets, we used an artificial validation set created by k-nn noise injection to control the training of MLPs [4, 11].

With $L$=5 classifiers and two classes, we have $m$=32 BKS cells. However, only eight cells were not empty and contained (224, 0), **(179, 204)**, (5, 0), **(196, 181)**, (1, 0), **(189, 202)**, **(206, 216)**, (0, 197) training vectors of the two classes, respectively. The fusion of the five classifiers with standard BKS method gave 37.00% of error on test data.

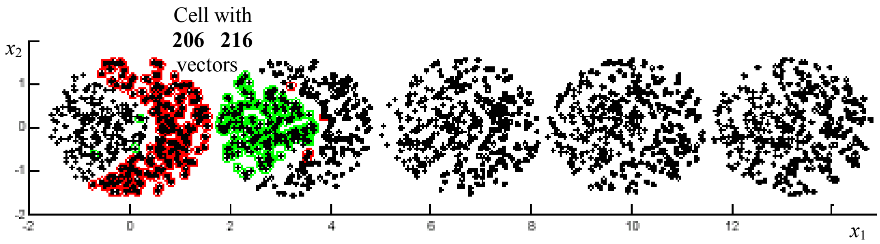


**Fig. 2.** Projection of test patterns of the artificial data set on the 2D informative feature space.

Then, we focused on the four ambiguous cells pointed out with bold characters. For example, Figure 2 points out test patterns falling in the **(206, 216)** cell in green and red colour. According to our strategy (Section 3), four single layer perceptrons were trained with patterns of each ambiguous cell, and then applied to test data falling in such cells. This allowed reducing the error to 4.95% (4.95% vs. 37.00%), that is a strong improvement of performances. Such result points out well the effectiveness of the proposed technique to increase BKS accuracy when ambiguous cells exist.

### 4.2 Experiments with the remote sensing data set

This data set consists of remote-sensing optical images [6]. Each pattern is characterised by an 8-dimensional feature vector. The data set consists of 15878 patterns belonging to two classes. We subdivided the data set into training set (4384+3555 patterns) and test set (4242+3606 patterns). The training set was sampled in order to create five disjoint subsets, that were used to perform ten independent trials. Each training subset was subdivided in four clusters by *k*-means clustering algorithm (*k*=4). Clustering was performed in the two-dimensional space obtained by principal component analysis. Four multiplayer perceptron (MLP) classifiers, with seven hidden nodes, were separately trained on each cluster by the Levenberg-Marquardt algorithm in order to obtain an ensemble of four specialized classifiers. The fusion of such four classifiers with standard BKS method provided 12.69% of error on test data. Then, six single layer perceptrons were trained with patterns of ambiguous cells, and applied to test data falling in such cells. We used single layer perceptrons only for ambiguous cells with a sufficient number of samples. This allowed reduce the error to 5.46% (5.46% vs. 12.69%). We did additional experiments using MLPs, with fifteen hidden units, trained on the entire training set (non specialised classifiers). In such case, the use of single layer perceptrons for ambiguous cells did not provide significant improvement, as such cells contained few patterns. Therefore, although definitive conclusions cannot be drawn on the basis of this limited set of experiments, the proposed technique seems to be effective for real data sets too, especially if ensembles made up of specialised classifiers are used.

## 5 Conclusions

The BKS method for fusing multiple classifiers was introduced in 1995 [1], and it is very popular in the pattern recognition literature. Reported experiments showed that it can provide good performances if large and representative data sets are available. Otherwise overfitting is likely to occur, and the generalization error quickly increases. In spite of this small sample size problem, analytical models of BKS generalization error are currently not available. BKS method also has an intrinsic limitation for the so called ambiguous cells, that is, for cells characterized by a low probability of the most representative class. So far the only strategy available to handle ambiguous cells is the reject option. In this paper, we proposed a simple analytical model that relates BKS generalization error to sample size for two-class cases. To the best of our knowledge, this is the first attempt to model BKS generalization error. We also discussed some implications of our model for BKS table design. Although this is a preliminary work limited to two class cases, such implications point out the practical relevance that the study of BKS generalization error can have. We also proposed to discriminate patterns falling in ambiguous BKS cells by using additional linear classifiers that work in the regions of the original feature space associated to such cells. Reported experiments showed that our technique could strongly improve BKS performances, especially when ensembles made up of specialised classifiers are used.

## References


[1] Huang, Y.S. and Suen, C.Y. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *IEEE Trans. On Pattern Analysis and Machine Intelligence* 17(1) (1995), pp. 90-94.

[2] C.Y.Suen, L.Lam. Multiple classifier combination methodologies for different output levels. Springer-Verlag Pub., *Lecture Notes in Computer Science*, Vol. 1857 (J.Kittler and F.Roli Eds., 2000), pp. 52-66.

[3] S.Raudys, Experts' boasting in trainable fusion rules, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, in press, 2003.

[4] F.Roli, S.Raudys, G. Marcialis, An experimental comparison of fixed and trained fusion rules for crisp classifiers outputs, Springer Pub., *Lecture Notes in Computer Science*, Vol. 2364 (F.Roli, J.Kittler Eds., 2002), pp. 232-241.

[5] C. K. Chow, On optimum error and reject tradeoff, *IEEE Trans. on Information Theory*, 16 (1970) 41-46.

[6] S.Raudys. Combining the expert networks: a review. *Proc. of the Int. Conference on Neural Networks and Artificial Intelligence*, (R. Sadykhov Ed.), 2-5 October 2001, Minsk, Belorus, pp. 81-91

[7] H.J.Kang, S.W.Lee, A dependency-based framework of combining multiple experts for the recognition of unconstrained handwritten numerals, *Proc. 1999 Int. Conf. On Computer Vision and Pattern Recognition*, Fort Collins, Colorado, USA, June 1999, pp. 124-129

[8] H.J.Kang, S.W.Lee, Combining classifiers based on minimization of a Bayes error rate, Proc. *5th int. Conf. On Document Analysis and Recognition*, Bangalore, India, 1999, pp. 398-401

[9] S.Raudys, *Statistical and Neural Classifiers: an integrated approach to design*, Springer, London, 2001

[10] G. Giacinto, F. Roli , Dynamic cassifier selection based on multiple classifier behaviour, *Pattern Recognition*, 34(9), 2001, pp. 179-181.

[11] M. Skurichina, S. Raudys, R.P.W. Duin. K-nearest neighbors directed noise injection in multilayer perceptron training, *IEEE Tr. on Neural Networks*, vol. 11, pp. 504–511, 2000.