Ensemble Approaches of Support Vector Machines for Multiclass Classification

Jun-Ki Min, Jin-Hyuk Hong, and Sung-Bae Cho

Department of Computer Science, Yonsei University
Biometrics Engineering Research Center

134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea

{loomlike, hjinh}@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Abstract. Support vector machine (SVM) which was originally designed for binary classification has achieved superior performance in various classification problems. In order to extend it to multiclass classification, one popular approach is to consider the problem as a collection of binary classification problems. Majority voting or winner-takes-all is then applied to combine those outputs, but it often causes problems to consider tie-breaks and tune the weights of individual classifiers. This paper presents two novel ensemble approaches: probabilistic ordering of one-vs-rest (OVR) SVMs with naïve Bayes classifier and multiple decision templates of OVR SVMs. Experiments with multiclass datasets have shown the usefulness of the ensemble methods.

Keywords: Support vector machines; Ensemble; Naïve Bayes; Multiple decision templates; Cancer classification, Fingerprint classification.

1 Introduction

Support Vector Machine (SVM) is a relatively new learning method which shows excellent performance in many pattern recognition applications [1]. It maps an input sample into a high dimensional feature space and tries to find an optimal hyperplane that minimizes the recognition error for the training data by using the non-linear transformation function [2]. Since the SVM was originally designed for binary classification, it is required to devise a multiclass SVM method [3].

Basically, there are two different trends for extending SVMs to multiclass problems. The first considers the multiclass problem directly as a generalization of the binary classification scheme [4]. This approach often leads to a complex optimization problem. The second decomposes a multiclass problem into multiple binary classification problems that can be solved by an SVM [5]. One-vs-rest (OVR) is a representative decomposition strategy, while winner-takes-all or error correcting codes (ECCs) is reconstruction schemes to combine the multiple outputs [6]. It has been pointed out that there is no guarantee on the decomposition-based approach to reach the optimal separation of samples [7]. There are several reasons for this, such as unbalanced sample sizes. However, these could be complemented by the proper selection of a model or a decision scheme.

In this paper, we present two novel ensemble approaches for applying OVR SVMs to multiclass classification. The first orders OVR SVMs probabilistically by using naïve Bayes (NB) classifier with respect to the subsumption architecture [8]. The latter uses multiple decision templates (MuDTs) of OVR SVMs [9]. It organizes the outputs of the SVMs with a decision profile as a matrix, and estimates the localized template from the profiles of the training set by using the K -means clustering algorithm. The profile of a test sample is then matched to the templates by a similarity measure. The approaches have been validated on the GCM cancer data [10] and the NIST4 fingerprints data [11]. Experimental results have shown the effectiveness of the presented methods.

2 Background

2.1 Multiclass Classification Using Binary SVMs

Since SVM is a basically binary classifier, a decomposition strategy for multiclass classification is required. As a representative scheme, the OVR strategy trains M (the number of classes) SVMs, where each SVM classifies samples into the corresponding class against all the others. The decision function of the j th SVM replaces the class label of the i th sample, c_i , with t_i as:

$$t_i = +1 \text{ if } c_i = j, \text{ otherwise } t_i = -1. \quad (1)$$

After constructing multiple SVMs, a fusion strategy is required to combine their outputs. Popular methods for combination such as winner-takes-all and ECCs [6] are widely employed. Winner-takes-all method classifies a sample into the class that receives the highest value among the L classifiers for the M -class problem. ECCs method generates a coding matrix $E \in \{-1, 0, 1\}^{M \times L}$ where $E_{i,j}$ represents an entry in the i th row and j th column of E . $E_{i,j} = -1$ (or 1) indicates that the points in class i are regarded as negative (or positive) examples when training the j th classifier. If $E_{i,j} = 0$, class i is not used when the j th classifier is trained. A test point is classified into the class whose row in the coding matrix has the minimum distance to the vector of the outputs of the classifiers.

2.2 Cancer Classification Based on Gene Expression Profiles

Recently developed DNA microarray technologies produce large volume of gene expression profiles and provide richer information on diseases. It simultaneously monitors the expression patterns of thousands of genes under a particular experimental environment. Especially the classification of cancers from gene expression profiles which commonly consists of feature selection and pattern classification has been actively investigated in bioinformatics [12]. Gene expression profiles provide useful information to classify different forms of cancers, but the data also include useless information for the classification. Therefore, it is important to find a small subset of genes sufficiently informative to distinguish cancers for diagnostic purposes [13].

2.3 Fingerprint Classification

Fingerprint classification is a technique that classifies fingerprints into a predefined category [14]. It is useful for an automated fingerprint identification system (AFIS) as a preliminary step of the matching process to reduce searching time. Henry system is the most widely used system for the fingerprint classification. It categorizes a fingerprint into one of five classes: whorl (W), right loop (R), left loop (L), arch (A), and tented arch (T) according to its global pattern of ridges. There are many different ways to extract and represent ridge information [15]. Especially, FingerCode proposed by Jain in 1999 [11] is a representative fingerprint feature extraction algorithm among them. This method tessellated a given fingerprint image into 48 sectors and transformed the image using the Gabor filters of four directions (0° , 45° , 90° , and 135°). Standard deviation was then computed on 48 sectors for each of the four transformed images in order to generate the 192-dimensional feature vector.

3 Novel Ensemble Methods for SVMs

3.1 PO-SVMs: Probabilistic Ordering of SVMs with Naïve Bayes Classifier

The method first estimates the probability of each class by using NB, and then organizes OVR SVMs as the subsumption architecture according the probability as shown in Fig. 1(a). Samples are sequentially evaluated with OVR SVMs. When a SVM is satisfied, the sample is classified into the corresponding class, while it is assigned with the class of the highest probability when no SVMs are satisfied. Fig. 1(b) shows the pseudo code of the proposed method.

The NB classifier estimates the posterior probability of each class given the observed attribute values for an instance, and the class with the highest posterior probability is finally selected [16]. In order to calculate the posterior probability, the classifier must be defined by the marginal probability distribution of variables and by a set of conditional probability distributions of each attribute A_i given each class c_j . They are estimated from the training set. When A_i is the i th state of the attribute A , which has a parent node B , and $count(A_i)$ is the frequency of cases in which the attribute A appears with the i th states, a priority probability $P(A_i)$ and a conditional probability $P(A_i|B_j)$ are estimated as follows:

$$P(A_i) = count(A_i)/n_T, \quad P(A_i | B_j) = count(A_i, B_j)/count(B_j). \quad (2)$$

Bayes' theorem yields the posterior probability of each class given n features as evidence over a class C and $F_1 \sim F_n$ by

$$P(C | F_1, \dots, F_n) = P(C)P(F_1, \dots, F_n | C)/P(F_1, \dots, F_n). \quad (3)$$

In practice, we are only interested in the numerator of the fraction, since the denominator does not affect C . Feature F_i is conditionally independent from the other feature F_j for $j \neq i$, so the probability of a class is calculated by equation (4):

$$P(C)P(F_1, \dots, F_n | C) = P(C)P(F_1 | C)P(F_2 | C) \dots P(F_n | C) = P(C) \prod_{i=1}^n P(F_i | C). \quad (4)$$

Since there will be a heavy computation when using high dimensional features for NB, we adopt different features from SVMs. In the following subsections, feature selection and extraction methods used for NB classifier are described.

Fig. 1. (a) Overview of the PO-SVMs and (b) its pseudo code

Feature Selection for Cancer Classification. A subset of informative genes is selected by using the feature selection process based on Pearson correlation. We define two ideal markers that represent a standard of good features, and utilize the features by scoring the respective similarity with each ideal marker. Two ideal markers are negatively correlated to represent two different aspects of classification boundaries. The first marker l_1 is high in class A and low in class B, while the second marker l_2 is low in class A and high in class B. Since this feature selection method is originally designed for binary classification, we select features based on the OVR scheme. Ten genes are selected for each class: the first five for the l_1 and the rest for the l_2 . When there are M classes, total $M \times 10$ genes are used to construct NB classifier.

The similarity between an ideal marker l and a gene g can be regarded as a distance, while the distance represents how far they are located from one another. A gene is regarded as an informative gene if the distance is small. Pearson correlation is used to measure the similarity as follows:

$$PC = \sum_{i=1}^n (l_i g_i) - \frac{\sum_{i=1}^n l_i \sum_{i=1}^n g_i}{n} \bigg/ \sqrt{\left(\sum_{i=1}^n l_i^2 - \left(\sum_{i=1}^n l_i \right)^2 / n \right) \times \left(\sum_{i=1}^n g_i^2 - \left(\sum_{i=1}^n g_i \right)^2 / n \right)} \quad (5)$$

Feature Extraction for Fingerprint Classification. Two representative features of fingerprints called the singular points (core and delta points) [14] and the pseudo ridge [17] are considered in order to construct a NB classifier. Fig. 2(a) shows an example. Locations and distances between the core C and other points are parameterized as shown in Fig. 2(b). If there are two core points, the nearest core to the center is denoted as C (if there is no core, C represents the center of the image). In this paper, as shown in Table 1, the number of cores and deltas, the location and distance between them, and the location and distance between cores and the end points of pseudo ridges are used for the NB classifier.

Fig. 2. (a) Fingerprint features which are used in this paper, (b) parameterized position and distance

Table 1. Features for the naïve Bayes classifier

Feature	Definition	State
Num_C, Num_D	Number of core points and delta points	0, 1, 2
Loc_{D1}, Loc_{D2}	Location of delta points ($D1$ and $D2$)	0, 1, 2, 3, 4, <i>absent</i>
Loc_{R1}, Loc_{R2}	Location of the end point of the pseudo ridge ($R1$ and $R2$)	0, 1, 2, 3, 4, <i>turn</i>
Dst_{D1}, Dst_{D2}	Distance between C and delta points	1, 2, 3, <i>absent</i>
Dst_{R1}, Dst_{R2}	Distance between C and the end point of the pseudo ridge	1, 2, 3, <i>turn</i>

3.2 MuDTs-SVMs: Multiple Decision Templates of SVMs

The original decision template (DT) is a classifier fusion method proposed by Kuncheva [18] which generates the template of each class (one per class) by averaging decision profiles for the training samples. The profile is a matrix that consists of the degree of support given by classifiers. The templates are estimated with the averaged profiles of the training data. In the test stage, the similarity between the decision profile of a test sample and each template is calculated. The class label refers to the class with the most similar templates.

The multiple decision templates (MuDTs) allow for clustering of each class and generate localized templates which are able to model intra-class variability and inter-class similarity of data. An overview of proposed method is shown in Fig. 3. For the M -class problem, a decision profile with the outputs of OVR SVMs is organized as represented in Eq (6), where $d_m(x_i)$ is the degree of support given by the m th classifier for the sample x_i .

$$DP(x_i) = \begin{bmatrix} d_1(x_i) \\ \vdots \\ d_M(x_i) \end{bmatrix}, \quad (6)$$

Since SVM is a binary classifier, we represented the profile to be one column matrix with positive or negative values. In order to generate multiple decision templates, profiles of training samples are clustered for each class. And then the localized template of the k th cluster in the class c , $DT_{c,k}$, is estimated by

$$DT_{c,k} = \frac{\sum_{i=1}^n ind_{c,k}(x_i) d_c(x_i)}{\sum_{i=1}^n ind_{c,k}(x_i)}. \quad (7)$$

An indicator function $ind_{c,k}(x_i)$ in Eq (7) refers to one if a sample x_i belongs to the k th cluster in the class c . If this is not the case, it refers to zero. In this paper, K -means algorithm, which is an iterative partitioning method that finds K compact clusters in the data using a minimum squared distance measures [19], was used for clustering method. In this paper, the number of clusters K is selected as $\sqrt{n/M}$, where n is the number of training samples and M is the number of classes.

Construct MuDTs-SVMs

Classify with MuDTs-SVMs

Notations

x_i : The i th sample of the training set
 $DT_{c,k}$: The decision template of cluster k in the class c
 $DP(x)$: The decision profile of the sample x
 $\text{Clustering}(c)$: decomposes the samples of the class c into K clusters
 $\text{Dist}(A, B)$: returns the distance between A and B

Construct MuDTs-SVMs

```

For ( $c = 0$  to  $M$ ) {
   $\text{Clustering}(c)$ 
  For ( $k = 0$  to  $K$ ) {
     $\text{count} = 0$ ,  $DT_{c,k} = [0, \dots, 0]$ 
    For ( $i = 0$  to  $N$ ) {
      If (The class of  $x_i = c$  and
        the cluster of  $x_i = k$ ) {
         $DT_{c,k} = DT_{c,k} + DP(x_i)$ 
        Increment  $\text{count}$ 
      }
    }
     $DT_{c,k} = DT_{c,k} / \text{count}$ 
  }
}
Return  $DT$ 

```

Classify with MuDTs-SVMs

```

 $\text{max} = 0$ 
For ( $c = 0$  to  $M$ ) {
  For ( $k = 0$  to  $K$ ) {
     $\text{distance} = \text{Dist}(DT_{c,k}, DP(x))$ 
    If ( $\text{distance} > \text{max}$ ) {
       $\text{max} = \text{distance}$ ,  $\text{label} = c$ 
    }
  }
}
Return  $\text{label}$ 

```

Fig. 3. (a) An overview of MuDTs method, (b) the pseudo code

In order to classify a new input sample, similarity between its profile and localized template of each cluster are calculated as:

$$dst_{c,k}(x) = |DT_{c,k} - U_c \cdot DP(x)|, \quad (8)$$

where

$$U_c = [u_1 \quad \cdots \quad u_M], \quad u_m = 1 \text{ if } m = c, \text{ otherwise } u_m = 0. \quad (9)$$

The sample is then classified into the class that contains the most similar cluster.

4 Experimental Results

We have verified the proposed methods on the GCM cancer data set [10] and the NIST4 fingerprint data set [11], summarized in Table 2.

Table 2. Summary of the datasets used

Data set	# Features	# Classes	# Samples	SVM kernel (parameter)
GCM	16,063 for SVMs 140 for NB	14	198	Linear
NIST4	192 for SVMs 10 for NB	5	3,937	Gaussian ($\sigma^2=0.0625$)

4.1 Experiments on GCM Cancer Data

GCM cancer dataset contains 144 training samples and 54 test samples with 16,063 gene expression levels. In this paper, all genes are used for SVMs and 140 genes are selected for NB based on Pearson correlation. There are 14 different tumor categories including breast adenocarcinoma (Br), prostate (Pr), lung adenocarcinoma (Lu), colorectal adenocarcinoma (Co), lymphoma (Ly), bladder (Bl), melanoma (Me), uterine adenocarcinoma (Ut), leukemia (Le), renal cell carcinoma (Re), pancreatic adenocarcinoma (Pa), ovarian adenocarcinoma (Ov), pleural mesothelioma (Pl), and central nervous system (Ce). Since the dataset provides only a few samples but lots of features, it is a challenging task for many machine learning researchers to construct a competitive classifier. Table 3 shows the performances of the previous works on this dataset.

Comparison results of the proposed method with several traditional approaches are presented in Table 4. OVA SVMs with the winner-takes-all strategy produced 77.8% classification accuracy, and NB yielded an accuracy of 74.8%, individually. PO-SVMs showed higher performance than the others with a classification accuracy of 81.5%, while MuDTs produced an accuracy of 77.8% (which is the same with that of winner-takes-all SVMs and ECCs). This may be because the clustering was not effective enough since there were only a few samples in the GCM dataset. The confusion matrices of proposed methods are presented in Table 5. Rows and columns denote the true class and the assigned class, respectively.

Table 3. Related works on the GCM cancer dataset

Author	Method	Accuracy (%)
Ramaswamy et al. (2001) [10]	OVR SVMs	77.8
Deutsch (2003) [20]	GA/SVMs	77.9
Li et al. (2004) [21]	SVMs random	63.3

Table 4. Classification accuracy for the GCM cancer dataset

Mehod	Product (SVMs+NB)	ECCs (SVMs)	DTs (SVMs)	PO-SVMs (SVMs+NB)	MuDTs-SVMs (SVMs)
Accuracy (%)	66.7	77.8	72.2	81.5	77.8

Table 5. Confusion matrices for the GCM cancer dataset (left: PO-SVMs, right: MuDTs-SVMs)

	Br	Pr	Lu	Co	Ly	Bl	Me	Ut	Le	Re	Pa	Ov	Pl	Ce
Br	3	0	0	0	0	0	0	0	0	0	1	0	0	0
Pr	0	5	0	0	0	0	0	0	0	0	0	1	0	0
Lu	0	0	4	0	0	0	0	0	0	0	0	0	0	0
Co	0	0	0	4	0	0	0	0	0	0	0	0	0	0
Ly	0	0	1	0	5	0	0	0	0	0	0	0	0	0
Bl	0	0	1	0	0	2	0	0	0	0	0	0	0	0
Me	0	0	0	0	0	0	1	0	0	0	1	0	0	0
Ut	0	0	0	0	0	0	0	2	0	0	0	0	0	0
Le	0	1	0	0	0	0	0	0	5	0	0	0	0	0
Re	0	0	0	0	0	0	0	1	0	2	0	0	0	0
Pa	0	0	0	1	0	1	0	0	0	0	1	0	0	0
Ov	0	0	0	1	0	0	1	0	0	0	0	2	0	0
Pl	0	0	0	0	0	0	0	0	0	0	0	0	3	0
Ce	0	0	0	0	0	0	0	0	0	0	0	0	0	4

	Br	Pr	Lu	Co	Ly	Bl	Me	Ut	Le	Re	Pa	Ov	Pl	Ce
Br	2	0	0	0	0	0	0	0	0	0	1	1	0	0
Pr	0	4	0	0	0	0	0	1	0	0	0	1	0	0
Lu	0	0	2	0	0	1	1	0	0	0	0	0	0	0
Co	0	0	0	4	0	0	0	0	0	0	0	0	0	0
Ly	0	0	0	0	6	0	0	0	0	0	0	0	0	0
Bl	0	0	0	0	0	2	1	0	0	0	0	0	0	0
Me	0	0	0	0	0	0	1	0	0	0	1	0	0	0
Ut	0	0	0	0	0	0	0	2	0	0	0	0	0	0
Le	0	1	0	0	0	0	0	0	5	0	0	0	0	0
Re	0	0	0	0	0	0	0	0	0	3	0	0	0	0
Pa	1	0	0	0	0	0	0	0	0	0	2	0	0	0
Ov	0	0	0	0	0	0	1	0	0	0	1	2	0	0
Pl	0	0	0	0	0	0	0	0	0	0	0	0	3	0
Ce	0	0	0	0	0	0	0	0	0	0	0	0	0	4

4.2 Experiments on NIST4 Fingerprint Data

This set consists of 4,000 scanned images (at 512×512 resolution) obtained from two impressions (F and S) of 2,000 fingerprints. Fingerprints were equally distributed into 5 classes, whorl (W), right loop (R), left loop (L), arch (A), and tented arch (T). Due to the ambiguity in fingerprint images, 350 fingerprints (17.5%) were cross-referenced with two classes. The first label was only considered in training phase while both labels were used in the test. In the experiment, the fingerprints of the first impression were used as the training set (F0001~F2000), and the other fingerprints constructed the test set. The FingerCode features was used for SVMs after normalization from -1 to 1 where some rejected images were included in the training set (1.4%) and the test set (1.8%) [11]. Singularities and pseudo ridge information was used for NB. Table 6 represents the performances of published fingerprint classification methods.

Several schemes for generating and combining multiple SVMs were compared with the proposed method (Table 7). OVA SVMs with the winner-takes-all method produced an accuracy of 90.1%, and NB yielded an accuracy of 85.4%, individually. Although the NB obtained lower accuracy than the SVMs, PO-SVMs (which combined NB and SVMs dynamically) showed an accuracy of 90.8% which is higher than the others.

Table 6. Related works on the NIST4 fingerprint dataset with FingerCode feature

Author	Method	Accuracy (%)
Jain et al. (1999) [11]	KNN+NN: Two stage classifiers	90.0
Yao et al. (2002) [22]	ECCs SVMs with RNN feature	90.0

Table 7. Classification accuracy for the NIST4 fingerprint dataset

Mehod	Product (SVMs+NB)	ECCs (SVMs)	DTs (SVMs)	PO-SVMs (SVMs+NB)	MuDTs-SVMs (SVMs)
Accuracy (%)	90.2	90.1	89.8	90.8	90.4

Table 8. Confusion matrices for the NIST4 fingerprint dataset (left: PO-SVMs, right: MuDTs-SVMs)

	W	R	L	A	T
W	373	10	10	0	0
R	4	374	0	6	15
L	5	0	377	8	9
A	0	6	4	365	40
T	1	8	15	39	295

	W	R	L	A	T
W	380	6	7	0	1
R	9	369	1	5	17
L	8	0	366	14	10
A	1	4	1	356	50
T	1	10	6	38	304

MuDTs-SVMs also achieved higher accuracy of 90.4% than the conventional methods. The confusion matrices of proposed methods are shown in Table 8.

5 Concluding Remarks

Multiclass classification is a challenging task in pattern recognition, where various approaches have been investigated especially using SVMs. Since SVM is a binary classifier, it is necessary to formulate decomposition and combination methods. In this paper, two novel ensemble approaches for OVR SVMs were proposed. The one uses probabilistic ordering of OVR SVMs with naïve Bayes classifier (PO-SVMs) and the other uses multiple decision templates of OVR SVMs (MuDTs-SVMs). Two benchmark data sets of GCM cancer data and NIST4 fingerprint data were used to verify these methods. As the future work, we will demonstrate the proposed methods with other popular benchmark datasets of multiclass, and study about the parameters for better performance.

Acknowledgments. This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei University.

References

1. Cortes, C., Vapnik, V.: Support-Vector Networks. *Machine Learning* 20(3), 273–297 (1995)
2. Cristianini, N., Shawe-Taylor, J.: *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge (2000)

3. Bredensteiner, E., Bennett, K.: Multicategory Classification by Support Vector Machines. *Computational Optimization and Applications*. 12(1), 53–79 (1999)
4. Arenas-Garcia, J., Perez-Cruz, F.: Multi-Class Support Vector Machines: A New Approach. In: 2003 IEEE Int. Conf. Acoustics, Speech, and Signal Processing, pp. II-781–784 (2003)
5. Hsu, C., Lin, C.: A Comparison of Methods for Multiclass Support Vector Machines. *IEEE Trans. Neural Networks*. 13(2), 415–425 (2002)
6. Rifkin, R.M., Klautau, A.: In Defense of One-Vs-All Classification. *J. Machine Learning Research*. 5, 101–141 (2004)
7. Lee, Y., Lin, Y., Wahba, G.: Multicategory Support Vector Machines. Tech. Rep. 1043, Dept. Statistics, Univ. of Wisconsin (2001)
8. Hong, J.-H., Cho, S.-B.: Multi-Class Cancer Classification with OVR-Support Vector Machines Selected by Naive Bayes Classifier. In: King, I., Wang, J., Chan, L., Wang, D. (eds.) *ICONIP 2006*. LNCS, vol. 4234, pp. 155–164. Springer, Heidelberg (2006)
9. Min, J.-K., Hong, J.-H., Cho, S.-B.: Effective Fingerprint Classification by Localized Models of Support Vector Machines. In: Zhang, D., Jain, A.K. (eds.) *Advances in Biometrics*. LNCS, vol. 3832, pp. 287–293. Springer, Heidelberg (2005)
10. Ramaswamy, S., et al.: Multiclass Cancer Diagnosis using Tumor Gene Expression Signatures. *Proc. National Academy of Science*. 98(26), 15149–15154 (2001)
11. Jain, A.K., Prabhakar, S., Hong, L.: A Multichannel Approach to Fingerprint Classification. *IEEE Trans. Pattern Analysis and Machine Intelligence*. 21(4), 348–359 (1999)
12. Cho, S.-B., Ryu, J.: Classifying Gene Expression Data of Cancer using Classifier Ensemble with Mutually Exclusive Features. *Proc. of the IEEE*. 90(11), 1744–1753 (2002)
13. Hong, J.-H., Cho, S.-B.: The Classification of Cancer based on DNA Microarray Data That Uses Diverse Ensemble Genetic Programming. *Artificial Intelligence in Medicine*. 36(1), 43–58 (2006)
14. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: *Handbook of Fingerprint Recognition*. Springer, Heidelberg (2003)
15. Yager, N., Amin, A.: Fingerprint Classification: A Review. *Pattern Analysis and Application*. 7(1), 77–93 (2004)
16. Liu, J., Li, B., Dillon, T.: An Improved Naïve Bayesian Classifier Technique Coupled with a Novel Input Solution Method. *IEEE Trans. Systems, Man, and Cybernetics-Part C: Applications and Reviews* 31(2), 249–256 (2001)
17. Zhang, Q., Yan, H.: Fingerprint Classification based on Extraction and Analysis of Singularities and Pseudo Ridges. *Pattern Recognition*. 37(11), 2233–2243 (2004)
18. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision Templates for Multiple Classifier Fusion: An Experimental Comparison. *Pattern Recognition*. 34(2), 299–314 (2001)
19. Jain, A.K., Dubes, R.C.: *Algorithms for Clustering Data*. Prentice-Hall, Englewood Cliffs (1988)
20. Deutsch, J.: Evolutionary Algorithms for Finding Optimal Gene Sets in Microarray Prediction. *Bioinformatics*. 19(1), 45–52 (2003)
21. Li, T., Zhang, C., Ogihara, M.: A Comparative Study of Feature Selection and Multiclass Classification Methods for Tissue Classification based on Gene Expression. *Bioinformatics*. 20(15), 2429–2437 (2004)
22. Yao, Y., Marcialis, G.L., Pontil, M., Frasconi, P., Roli, F.: Combining Flat and Structured Representations for Fingerprint Classification with Recursive Neural Networks and Support Vector Machines. *Pattern Recognition*. 36(2), 397–406 (2003)