On a New Measure of Classifier Competence Applied to the Design of Multiclassifier Systems

Tomasz Woloszynski and Marek Kurzynski

Wroclaw University of Technology, Chair of Systems and Computer Networks,
Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland
marek.kurzynski@pwr.wroc.pl

Abstract. This paper presents a new method for calculating competence of a classifier in the feature space. The idea is based on relating the response of the classifier with the response obtained by a random guessing. The measure of competence reflects this relation and rates the classifier with respect to the random guessing in a continuous manner. Two multiclassifier systems representing fusion and selection strategies were developed using proposed measure of competence. The performance of multiclassifiers was evaluated using five benchmark databases from the UCI Machine Learning Repository and Ludmila Kuncheva Collection. Classification results obtained for three simple fusion methods and one multiclassifier system with selection strategy were used for a comparison. The experimental results showed that, regardless of the strategy used by the multiclassifier system, the classification accuracy has increased when the measure of competence was employed. The improvement was most significant for simple fusion methods (*sum, product* and *majority vote*). For all databases, two developed multiclassifier systems produced the best classification scores.

1 Introduction

One of the most important tasks in optimizing a multiclassifier system is to select a group of competent (adequate) classifiers from a pool of classifiers. There are two main approaches to this problem: static selection scheme in which ensamble of classifiers is selected for all test patterns, and dynamic selection method which explores the use of different classifiers for different test patterns [6]. The most dynamic classifier selection schemes use the concept of classifier "competence" on a defined neighbourhood or region [7], such as the local accuracy *a priori* [11] or *a posteriori* methods [2] or overall local accuracy and local class accuracy [10].

In this paper we present a new method of dynamic optimization scheme of multiclasifier system based on a class-independent measure of classifier competence in the feature space. The value of the proposed measure of competence is calculated with respect to the response obtained by random guessing. In this way it is possible to evaluate a group of classifiers against a common reference point. Competent (incompetent) classifiers gain with such approach meaningful interpretation, i.e. they are more (less) accurate than the random classifier.

P. Foggia, C. Sansone, and M. Vento (Eds.): ICIAP 2009, LNCS 5716, pp. 995–1004, 2009.

This paper is divided into five sections and organized as follows. In Section 2 the measure of classifier competence is presented and three different propositions of its functional form are given. In Section 3 two multiclassifier systems based on selection strategy and fusion strategy are developed. Computer experiments are described in Section 4 and Section 5 concludes the paper.

2 The Measure of Classifier Competence

Consider an n-dimensional feature space $\mathcal{X} \subseteq \mathcal{R}^n$ and a finite set of class labels $\mathcal{M} = \{1, 2, \ldots, M\}$. Let

$$\psi : \mathcal{X} \to \mathcal{M} \tag{1}$$

be a classifier which produces a set of discriminant functions $(d_1(x), d_2(x), \ldots, d_M(x))$ for a given object described by a feature vector x. The value of the discriminant function $d_i(x),\ i = 1, 2, \ldots, M$ represents a support given by the classifier ψ for the i-th class. Without loss of generality we assume that $d_i(x) > 0$ and $\sum d_i(x) = 1$. Classification is made according to the maximum rule, i.e.

$$\psi(x) = i \Leftrightarrow d_i(x) = \max_{k \in \mathcal{M}} d_k(x). \tag{2}$$

We assume that, apart from a training and testing datasets, a validation dataset is also available. The validation dataset is given as

$$V_N = \{(x_1, i_1), (x_2, i_2), \ldots, (x_N, i_N)\}, \tag{3}$$

where $x_k \in \mathcal{X},\ k = 1, 2, \ldots, N$ denotes the feature vector representing the k-th object in the dataset and $i_k \in \mathcal{M}$ denotes the object's class label.

The set (3) can be applied to the evaluation of classifier competence, i.e. its capability to correct activity (correct classification) in the whole feature space. For this purpose the *potential function* model will be used [7], [9]. In this approach feature space is considered as a "competence field" which is determined by the sources of competence located at the points $x_k,\ k = 1, 2, \ldots, N$.

We define the source competence $K_\psi(x_k)$ of the classifier ψ at a point $x_k \in \mathcal{X}$ from the set (3) as a function of class number M and the support of correct class $d_{i_k}(x_k)$ having the following properties:

1. $K_\psi(x_k)$ is strictly increasing function of $d_{i_k}(x_k)$ (for any M),
2. $K_\psi(x_k) = -1$ for $d_{i_k}(x_k) = 0$,
3. $K_\psi(x_k) = 0$ for $d_{i_k}(x_k) = 1/M$,
4. $K_\psi(x_k) = 1$ for $d_{i_k}(x_k) = 1$.

The idea of function K_ψ is based on relating the response of the classifier ψ with the response obtained by a random guessing. The source competence reflects this relation and rates the classifier with respect to the random guessing in a continuous manner. If the support for the correct class is lower than the probability of random guessing, then the source competence is negative and ψ is evaluated as an incompetent classifier. If in turn this support is greater than the probability

Table 1. The overview of the cases of the source competence

Support for the correct class	The source competence	Evaluation of the classifier ψ
$d_{i_k}(x_k) = 1$	$K_\psi(x_k) = 1$	The classifier is absolutely competent
$1 > d_{i_k}(x_k) > \frac{1}{M}$	$1 > K_\psi(x_k) > 0$	The classifier is competent
$d_{i_k}(x_k) = \frac{1}{M}$	$K_\psi(x_k) = 0$	The classifier is neutral (equivalent to the random guessing)
$\frac{1}{M} > d_{i_k}(x_k) > 0$	$0 > K_\psi(x_k) > -1$	The classifier is incompetent
$d_{i_k}(x_k) = 0$	$K_\psi(x_k) = -1$	The classifier is absolutely incompetent

of random guessing then K_ψ is positive and ψ is regarded as a competent classifier. The overview of different cases of values $d_{i_k}(x_k)$ and $K_\psi(x_k)$ and related interpretation of classifier competence at the point x_k is presented in Table 1.

The following functions will be considered as the source competence:

1. Logarithmic function:

$$K_\psi^{(1)}(x_k) = \begin{cases} 2\, d_{i_k}(x_k) - 1 & \text{for } M = 2, \\ \frac{log[M(M-2)\, d_{i_k}(x_k)+1]}{log(M-1)} - 1 & \text{for } M > 2. \end{cases} \tag{4}$$

2. Exponential function:

$$K_\psi^{(2)}(x_k) = 1 - 2^{1-\frac{(M-1)d_{i_k}(x_k)}{1-d_{i_k}(x_k)}} \tag{5}$$

3. Piecewise linear function:

$$K_\psi^{(3)}(x_k) = \begin{cases} \frac{M}{M-1}\, d_{i_k}(x_k) - \frac{1}{M-1} & \text{for } \frac{1}{M} \le d_{i_k}(x_k) \le 1 \\ M\, d_{i_k}(x_k) - 1 & \text{for } 0 \le d_{i_k}(x_k) \le \frac{1}{M}. \end{cases} \tag{6}$$

The source competence also depends on the number of classes in the classification problem. This dependence is shown in Fig.1 for functions (4), (5) and (6).

The competence of the classifier ψ at any given point x is defined as the weighted sum of source competences $K_\psi(x_k)$, $k = 1, 2, \ldots, N$ with weights exponentially dependent on the distance $\|x - x_i\|$ between points x_k and x, namely:

$$C_\psi(x) = \sum_{k=1}^{N} K_\psi(x_k)\ \exp[-\|x - x_i\|]. \tag{7}$$

Classifier ψ is competent in the given $x \in \mathcal{X}$ if $C_\psi(x) > 0$, otherwise ψ is regarded as an incompetent classifier.

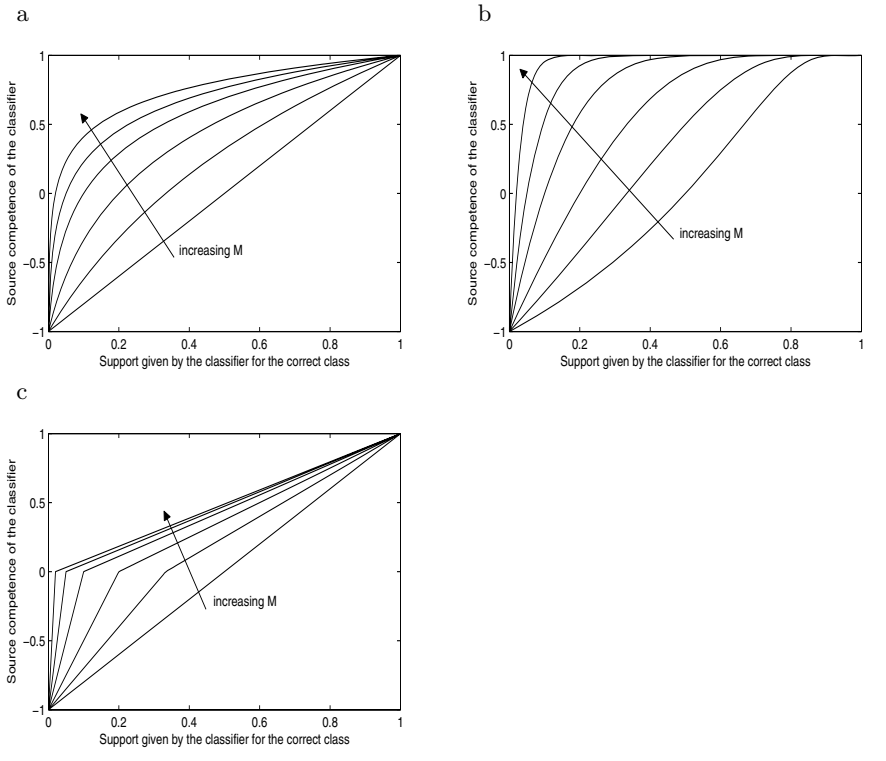

Fig. 1. The source competence $K^{(1)}$(a), $K^{(2)}$ (b) and $K^{(3)}$ (c) plotted against the support for the correct class for different number of classes ($M = 2, 3, 5, 10, 20, 50$)

Although any given metric can be used in the definition of the distance $\|x - x_i\|$ we propose a modified Euclidean distance in the following form:

$$\|x - x_k\| = \frac{0.5}{h_{opt}^2}(x - x_k)^T(x - x_k), \tag{8}$$

where the parameter h_{opt} is the optimal smoothing parameter obtained for the validation dataset V_N by the Parzen kernel estimation. The kernel estimation was used to normalize ranges of features and to ensure that each source competence will affect only its neighbourhood. Note that in this step no Parzen classifier is needed, i.e. only the value of the optimal smoothing parameter is calculated.

3 Application to Multiclassifier Systems

The measure of competence can be incorporated in virtually any multiclassifier system providing that $\mathcal{X}$ is an metric space. In this chapter we describe

two multiclassifier systems based on proposed measure of competence, each one employing different strategy.

Let us assume that we are given a set (pool) of trained base classifiers $\mathcal{L} = \{\psi_1, \psi_2, \ldots, \psi_L\}$ and the validation dataset V_N. We define the multiclassifier $F_1(x)$ to be the classifier with the highest positive competence value at the point x:

$$F_1(x) = \psi_l(x) \;\Leftrightarrow\; C_{\psi_l}(x) > 0 \wedge C_{\psi_l}(x) = \max_{k=1,2,\ldots,L} C_{\psi_k}(x). \tag{9}$$

The multiclassifier F_1 uses a selection strategy, i.e. for each object described by a feature vector x it selects a single classifier to be used for classification. In the case where all classifiers have negative values of competence classification is made according to the random classifier. The random classifier draws a class label using a discrete uniform distribution with probability value $\frac{1}{M}$ for each class.

The multiclassifier F_2 represents a fusion approach where the final classification is based on responses given by all competent classifiers:

$$F_2(x) = \sum_{l \in L_{pos}} C_{\psi_l}(x)\psi_l(x), \tag{10}$$

where the set L_{pos} contains indices of classifiers with positive values of competence. Again, the random classifier is used in the case where all classifiers have negative values of competence. The classification is made according to the maximum rule given in (2).

4 Experiments

4.1 Benchmark Data and Experimental Setup

Benchmark databases used in the experiments were obtained from the UCI Machine Learning Repository [1] (*Glass*, *Image segmentation*, *Wine*) and Ludmila Kuncheva Collection [8] (*Laryngeal3* and *Thyroid*). Selected databases represent classification problems with objects described by continuous feature vectors. For each database, feature vectors were normalized for zero mean and unit standard deviation (SD). Three datasets were generated from each database, i.e. training, validation and testing dataset. The training dataset was used to train the base classifiers. The values of the competence for each base classifier were calculated using the validation dataset. The testing dataset was used to evaluate the accuracy of tested classification methods. A brief description of each database is given in Table 2.

For each database, 30 trials with the same settings were conducted. The accuracy of each classifier and multiclassifier used was calculated as the mean (SD) value of these 30 trials. In this way it was possible to evaluate both the accuracy and the stability of examined multiclassifier systems.

Table 2. A brief description of each database; *%training*, *%validation* and *%testing* indicate the percentage of objects used for generation of the training, validation and testing dataset, respectively

Database	#classes	#objects	#features	%training	%validation	%testing
Glass	6	214	9	30	40	100
Image segm.	7	2310	19	20	30	100
Wine	3	178	13	20	40	100
Laryngeal3	3	353	16	20	40	100
Thyroid	3	215	5	20	40	100

4.2 Classifiers

The following set of base classifiers was used in the experiments [3]:

1. LDC - linear discriminant classifier based on normal distributions with the same covariance matrix for each class;
2. QDC - quadratic discriminant classifier based on normal distributions with different covariance matrix for each class;
3. NMC - nearest mean classifier;
4. 1-NN - nearest neighbour classifier;
5. 5-NN - k-nearest neighbours classifier with $k = 5$;
6. 15-NN - k-nearest neighbours classifier with $k = 15$;
7. PARZEN1 - Parzen classifier with the Gaussian kernel and optimal smoothing parameter h_{opt};
8. PARZEN2 - Parzen classifier with the Gaussian kernel and smoothing parameter $h_{opt}/2$;
9. TREE - Tree classifier with Gini splitting criterion and pruning level set to 3;
10. BPNN1 - Feed-forward backpropagation neural network classifier with two hidden layers (2 neurons in each layer) and the maximum number of epochs set to 50;
11. BPNN2 - Feed-forward backpropagation neural network classifier with one hidden layer (5 neurons) and the maximum number of epochs set to 50.

Proposed multiclassifier systems were compared against a classifier selection method with class-independent competence estimation of each classifier (CS-DEC) [9]. This method was chosen because it is similar to the method presented in this paper, i.e. CS-DEC evaluates the competence of a classifier in a continuous manner with distance defined as a potential function. Although other approach is commonly used [2,10] (i.e. evaluation of competence using k-NN neighbourhood instead of potential functions), currently it is not known if these two methods produce significantly different results [7].

For a better evaluation of differences between proposed multiclassifiers and the method used for a comparison, 95% confidence intervals (CI) of accuracy were calculated. The intervals obtained for developed multiclassifiers were compared

against the interval calculated for the CS-DEC. The accuracies of a given multiclassifier and CS-DEC were considered statistically significant if their respective 95% CI intervals were not overlapping.

The performances of four groups of simple fusion methods were also evaluated. The group A contained the *sum, product* and *majority vote* fusion methods used with all base classifiers. The groups B, C and D contained the same three fusion methods with the exception that only the base classifiers with positive value of competence $C_\psi(x)$ were used. This competence was calculated according to the source competence $K^{(1)}$ (group B), $K^{(2)}$ (group C) and $K^{(3)}$ (group D). If no competent base classifier for a given object x was available, the classification was made using the random classifier.

4.3 Results and Discussion

The results obtained for the base classifiers are shown in Table 3. It can be seen from the table that the set of base classifiers provided diversity needed in the multiclassifier systems, i.e. there was no single superior classifier and the range of classification scores was large (high values of the SD). The best overall accuracy averaged over all databases was achieved by the nearest neighbour classifier 1-NN (85.9%), followed shortly by two Parzen classifiers: PARZEN2 (85.8%) and PARZEN1 (85.2%). The lowest averaged classification scores were obtained for the quadratic discriminant classifier QDC (50.9%) and the neural network classifier BPNN1 (61.8%). The relatively low accuracy of the neural network classifier can be explained by the fact, that the learning process was stopped after just 50 epochs.

The results obtained for the multiclassifier systems are presented in Table 4. It can be noticed from the table that the group of fusion methods which used all base classifiers (group A) achieved the lowest classification accuracies. This indicates that weak classifiers from the pool can noticeably affect the *sum, product* and *majority vote* fusion methods. However, the same fusion methods combined

Table 3. The results obtained for the base classifiers. The best score for each database is highlighted.

	Database / Mean (SD) accuracy [%]				
Classifier	Glass	Image seg.	Laryngeal3	Thyroid	Wine
LDC	61.4(3.5)	90.8(0.7)	70.7(2.0)	91.7(2.3)	**95.9(1.4)**
QDC	27.1(10.8)	80.6(6.1)	31.5(15.6)	63.2(34.4)	52.1(20.1)
NMC	46.1(5.8)	84.2(1.1)	67.0(4.1)	92.7(2.7)	95.4(1.3)
1-NN	**76.0(2.1)**	**93.6(0.5)**	72.0(2.5)	93.6(1.9)	94.5(1.8)
5-NN	64.7(2.4)	90.3(1.0)	71.9(1.8)	88.0(4.4)	94.4(2.3)
15-NN	51.6(5.6)	85.8(1.0)	70.3(3.2)	73.3(3.6)	87.8(9.2)
PARZEN1	70.7(3.5)	93.2(0.7)	**75.1(1.9)**	92.1(2.9)	95.1(1.8)
PARZEN2	75.1(2.2)	93.1(0.5)	72.6(2.5)	**93.7(2.0)**	94.5(1.8)
TREE	60.2(5.1)	81.8(3.6)	67.5(3.7)	85.6(5.8)	81.1(5.7)
BPNN1	45.6(7.3)	40.8(5.3)	65.4(4.3)	85.5(7.3)	71.4(22.2)
BPNN2	48.7(8.0)	40.9(5.2)	66.6(4.4)	89.2(6.6)	83.2(13.4)

Table 4. The results obtained for the mutliclassifier systems (description in the text). The best score for each database is highlighted. Asterisks denote statistically significant differences with respect to the CS-DEC method.

	Database / Mean (SD) accuracy [%]				
Multiclassifier	Glass	Image seg.	Laryngeal3	Thyroid	Wine
Sum^A	41.9(15.3)	93.3(1.8)	73.5(9.2)	91.8(5.1)	96.3(1.4)
Product^A	28.9(8.8)	91.7(2.2)	43.0(15.7)	90.5(7.2)	83.3(9.4)
Majority vote^A	68.7(3.5)	93.5(0.9)	74.9(1.3)	92.5(2.8)	95.9(1.4)
Sum^B	74.3(2.6)	95.6(0.8)	81.6(1.4)	95.1(1.8)	97.8(1.1)
Product^B	74.3(2.4)	95.7(0.7)	80.8(1.5)	95.3(1.5)	96.5(2.4)
Majority vote^B	72.1(3.1)	94.1(0.9)	79.3(1.6)	94.2(2.0)	96.9(1.5)
Sum^C	74.3(2.5)	95.5(0.7)	81.3(1.4)	95.0(1.9)	97.8(1.1)
Product^C	74.2(2.4)	95.5(0.6)	80.7(1.4)	95.2(1.7)	96.6(2.5)
Majority vote^C	72.1(3.1)	94.1(0.9)	79.2(1.5)	94.2(2.2)	96.9(1.5)
Sum^D	74.3(2.6)	95.7(0.8)	81.6(1.6)	95.2(1.7)	97.8(1.1)
Product^D	74.3(2.5)	95.7(0.7)	80.7(1.6)	95.3(1.6)	96.5(2.4)
Majority vote^D	72.4(2.9)	94.1(1.0)	79.5(1.6)	94.4(2.0)	96.9(1.5)
CS-DEC	78.0(2.0)	94.6(0.4)	80.7(1.7)	95.5(1.8)	97.0(1.3)
$F_1^{(1)}$	**78.1(2.1)**	95.7(0.5)*	81.2(1.6)*	95.9(1.6)*	97.2(1.1)
$F_1^{(2)}$	76.7(2.6)*	95.7(0.5)*	81.0(1.5)	**96.0(1.5)***	97.4(1.2)*
$F_1^{(3)}$	**78.1(1.9)**	95.7(0.5)*	81.2(1.7)*	**96.0(1.5)***	97.1(1.1)
$F_2^{(1)}$	76.4(2.5)*	**95.9(0.6)***	**82.0(1.6)***	95.7(1.8)	97.8(1.0)*
$F_2^{(2)}$	76.0(2.3)*	**95.9(0.7)***	81.8(1.5)*	95.6(1.8)	**97.9(1.0)***
$F_2^{(3)}$	76.7(2.3)*	**95.9(0.6)***	**82.0(1.7)***	95.9(1.6)*	97.8(1.0)*

with the measure of competence (group B, C and D) produced the classification scores which were, on average, 10% higher (e.g. in the case of *product* multiclassifier and Glass database the improvement was over 45%). This can be explained by the fact, that for each input object x only classifiers that are assumingly more accurate than the random guessing were used in the classification process. It can be shown that a set of weak base classifiers, where each classifier performs just slightly better than the random classifier, can be turned into a powerful classification method. Such approach has been successfully used in boosting algorithms [4].

The multiclassifier CS-DEC used for comparison produced better classification scores averaged over all databases (89.1%) than the fusion methods from groups A — D. However, it was outperformed by two developed multiclassifier systems F_1 (89.6%) and F_2 (89.7%). The multiclassifiers F_1 and F_2 produced the best stability (the SD value of 1.4% averaged over all databases), followed by the CS-DEC and the group B and $Csum$ multiclassifiers (all 1.5%). Results obtained indicate that proposed measure of competence produced accurate and reliable evaluations of the base classifiers over all feature space. This in turn enabled multiclassifier systems to perform equally well for all input objects x. This can be explained by the fact, that the set of base classifiers was diversified, i.e. for each database and each object x to be classified, at least one competent

classifier was always available (the random classifier in F_1 and F_2 methods was never used).

Statistically significant differences were obtained in 23 out of 30 cases (5 databases times 6 developed multiclassifiers). Four times these differences favored CS-DEC method (Glass database only). This shows that proposed multiclassifiers indeed display improvement in the class-independent competence measure over currently used approach.

It is interesting to note that results for all three forms of the source competence are almost identical, what suggests that character of dependence K_ψ on support d_{i_k} is rather negligible.

5 Conclusions

A new method for calculating the competence of a classifier in the feature space was presented. Two multiclassifier systems incorporating the competence were evaluated using five databases from the UCI Machine Learning Repository and Ludmila Kuncheva Collection. The results obtained indicate that the proposed measure of competence can eliminate weak (worse than random guessing) classifiers from the classification process. At the same time strong (competent) classifiers were selected in such a way that the final classification accuracy was always better than the single best classifier from the pool.

Simple fusion methods (*sum, product* and *majority vote*) displayed the greatest improvement when combined with the measure of competence. Two developed multiclassifier systems based on selection strategy (F_1) and fusion strategy (F_2) achieved both the best classification scores and stability, and outperformed other classification methods used for comparison. Experimental results showed that the idea of calculating the competence of a classifier by relating its response to the response obtained by the random guessing is correct, i.e. a group of competent classifiers provided better classification accuracy than any of the base classifiers regardless of the strategy which was employed in the multiclassifier system.

References

1. Asuncion, A., Newman, D.: UCI Machine Learning Repository. University of California, Department of Information and Computer Science, Irvine, CA (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
2. Didaci, L., Giacinto, G., Roli, F., Arcialis, G.: A study on the performance of dynamic classifier selection based on local accuracy estimation. Pattern Recognition 38, 2188–2191 (2005)
3. Duda, R., Hart, P., Stork, D.: Pattern Classification. Wiley-Interscience, Hoboken (2001)
4. Freund, Y., Schapire, R.: Experiments with a new boosting algorithm. In: Machine Learning: Proceedings of the Thirteenth International Conference, pp. 148–156 (1996)
5. Giacinto, G., Roli, F.: Design of effective neural network ensembles for image classification processes. Image Vision and Computing Journal 19, 699–707 (2001)

6. Ko, A., Sabourin, R., Britto, A.: From dynamic classifier selection to dynamic ensamble selection. Pattern Recognition 41, 1718–1733 (2008)
7. Kuncheva, L.: Combining Pattern Classifiers: Methods and Algorithms. Wiley-Interscience, New Jersey (2004)
8. Kuncheva, L.: Collection, http://www.informatics.bangor.ac.uk/ kuncheva/activities/real_data_full_set.htm
9. Rastrigin, L.A., Erenstein, R.H.: Method of Collective Recognition. Energoizdat, Moscow (1981)
10. Woods, K., Kegelmeyer, W.P., Bowyer, K.: Combination of multiple classifiers using local accuracy estimates. IEEE Transactions on Pattern Analysis and Machine Intelligence 19, 405–410 (1997)
11. Woloszynski, T., Kurzynski, M.: On a new measure of classifier competence in the feature space. Computer Recognition Systems 3 (2009) (in print)