Methods for Dynamic Classifier Selection

Giorgio Giacinto and Fabio Roli
Dept. of Electrical and Electronic Eng., University of Cagliari, Italy
Piazza D'Armi, 09123, Cagliari, Italy
Phone: +39-070-6755874 - Fax: +39-070-6755900 - e-mail {giacinto, roli}@diee.unica.it

Abstract

In the field of pattern recognition, the concept of Multiple Classifier Systems (MCSs) was proposed as a method for the development of high performance classification systems. At present, the common "operation" mechanism of MCSs is the "combination" of classifiers outputs. Recently, some researchers pointed out the potentialities of "dynamic classifier selection" as a new operation mechanism. In a previous paper, the authors discussed the advantages of "selection-based" MCSs and proposed an algorithm for dynamic classifier selection [1]. In this paper, a theoretical framework for dynamic classifier selection is described and two methods for selecting classifiers are proposed. Reported results on the classification of different data sets show that dynamic classifier selection is an effective method for the development of MCSs.

1. Introduction

In the literature, many combination-based MCSs have been described [2-5]. The most of the combination methods used in such systems assume that classifiers forming the MCS make "independent" classification errors. This assumption is necessary to guarantee an increase of classification accuracy with respect to the accuracies provided by individual classifiers [5-8].

Unfortunately, in real pattern recognition applications, it is usually difficult to design a set of classifiers that satisfies the above assumption of independent errors. Consequently, some papers described methods to design a set of "uncorrelated" classifiers [9, 10], or combination functions that do not need of the independence assumption [11, 12].

Recently, some researchers proposed a different approach to the development of MCSs based on the concept of dynamic classifier selection [1, 4, 13]. (In a previous paper, the authors referred to such MCSs as "selection-based" MCSs [1]). Roughly speaking, selection-based MCSs are based on a function that, for each test pattern, dynamically select the classifier that correctly classify it. The authors pointed out that selection-based MCSs, as compared with the combination-based ones, do not need of the assumption of "independent" classifiers [1]. For each test pattern, selection-based MCSs need just one classifier that correctly classifies it. It is easy to see that this assumption is much more easy to be satisfied than the independence assumption. Therefore, the choice of the classifiers forming a MCS is much more simple for a selection-based MCS than for one based on combination. In addition, it is also easy to see that an "optimal" selector, that is, an "oracle", should be preferred to any "optimal" combination method [1, 4, 13].

This paper opens with a short overview on the related work (Section 2). Section 3.1 describes a theoretical framework for dynamic classifier selection. This framework is aimed to show that dynamic classifier selection is a method for designing an optimal Bayesian classifier. Afterwards, two selection functions and an algorithm based on them are described (Sections 3.2, 3.3, and 3.4). Experimental results and comparisons are reported in Section 4. Conclusions are drawn in Section 5.

2. Related work

First of all, it is necessary to highlight that the concept of dynamic classifier selection is currently used in the fields of artificial neural networks and pattern recognition with two different meanings.

In the field of artificial neural networks, the concept of dynamic classifier selection is used for the development of the so called "modular" approaches to the combination of neural nets [7]. Modular approaches imply an ensemble of "specialized" nets, where each net is responsible for some aspect of the classification task. Therefore, all the nets are necessary to solve the whole task. On the other hand, in the field of pattern recognition, the concept of dynamic classifier selection is currently used to develop MCSs where each classifier is able to solve the whole classification task. It should be quite easy to see the basic differences among the two uses of the concept of dynamic selection. Modular approaches focus on "task decomposition", while selection-based MCSs focus on "dynamic complementarities" of the classifiers with respect to the whole classification task.

According to the above remark, a brief overview of related work carried out in the field of pattern recognition is given in this section. First of all, it is worth noticing that, to the best of our knowledge, very few papers have addressed the topic of selection-based MCSs.

Srihari et al. were the first to introduce the concept of dynamic classifier selection as an alternative to combination in MCSs [4]. They briefly outlined the potentialities of this concept and proposed a selection method based on a partition of the training set according to the state of agreement of the classifiers on the "top choices" (a class ranking was assumed). A different combination function was then dynamically selected according to such a state of agreement (see page 71 in [4] for further details).

Afterwards, the authors and Woods et al. proposed some algorithms for dynamic classifier selection [1, 13]. Although developed independently, these algorithms are based on the same concept of "local accuracy estimates". In the following, the main differences will be highlighted.

Woods et al. proposed a method for dynamic classifier selection based on classifier local accuracy estimates. For each classifier, an estimate of accuracy in local regions of feature space surrounding a test sample is computed. Local regions are defined in terms of the k-nearest neighbors in the training data. They proposed two methods for estimating local accuracy. One, named "overall local accuracy", is simply the percentage of training samples in the local region that are correctly classified. The other method ("local class accuracy") takes into account the data class assigned by a classifier to the test sample and then computes the percentage of the local training samples correctly assigned to that class.

Although based on the same concept of "local accuracy", the method proposed by the authors in [1] exhibited the following main differences:

- local accuracies were estimated using the class posterior probabilities, while Woods et al. simply considered the "labels" assigned to the training samples (i.e., the "hard" classifications);
- the distances of the training samples, belonging to local regions, from the test sample were computed and used in the estimates of local accuracies. This allowed us to handle more effectively the "key" problem related to the choice of the size of the "local region" and to provide estimates of local accuracies more "robust";
- the classifier selection algorithm computed a sort of "confidence" for the selection and, consequently, "rejection" was possible;
- a simple method for designing selection-based MCSs was proposed. Roughly speaking, this method subdivided the training set into "partitions" and assigned one classifier to each partition.

3. Methods for dynamic classifier selection

3.1 A Theoretical framework for dynamic classifier selection

Let us consider a classification task for M data classes. Each class is assumed to represent a set of specific patterns, each pattern being characterized by a feature vector **X**. Let $P(\omega_i)$ be the a priori probability for the class ω_i, i = 1,..,M, and let $p(\mathbf{X}|\omega_i)$ be the conditional probability density function for the patterns belonging to the class ω_i. Any classification algorithm subdivides the feature space into M "decision regions", R_i, i = 1,..,M, so that the patterns belonging to the decision region R_i are assigned to the class ω_i. It is well known that the probability of correct classification can be computed by $\sum_{i=1}^{M}\int_{R_i} p(\mathbf{X}\,|\,\omega_i)P(\omega_i)d\mathbf{X}$. This probability is maximum when the decision regions R_i are defined in order to maximize the integrands. This happens if patterns are assigned to the M classes according to the Bayes decision theory, that is, each pattern **X** is assigned to the class for which the $P(\omega_i|\mathbf{X})$ is maximum. Let us refer to such regions R_i as "Bayesian decision regions".

In real applications, it is very difficult to estimate exactly the class posterior probabilities, and, consequently, classification algorithms provide decision regions that do not correspond with the Bayesian ones.

Let us assume that K different classifiers are available to solve the classification task at hand. For each classifier C_j, j = 1,..,K, let $R_i^{(j)}$, i = 1,..,M, be the related decision regions.

Without loosing in generality, each region $R_i^{(j)}$ can be regarded as subdivided into two subregions $R_i^{j+} = R_i^j \cap R_i^B$ and $R_i^{j-} = R_i^j - R_i^{j+}$. It is easy to see that each classifier C_j corresponds to an "optimal" Bayesian classifier in the regions $R_i^{(j)+}$, i = 1,..,M, while it takes non Bayesian decisions into the regions $R_i^{(j)-}$.

Let us hypothesize that the Bayesian regions R_i can be "recovered" as follows:

$$R_i = \bigcup_{j=1}^{K} R_i^{(j)+} \qquad (1)$$

It is easy to see that the above assumption requires K "specialized" classifiers, that is, classifiers that exhibit a high probability of correct classification in different regions of the feature space.

According to the hypothesis in equation (1), an optimal Bayesian classifier can be obtained by "selecting" the classifier C_j, j = 1,..,K, just for the patterns belonging to the regions $R_i^{(j)+}$, i = 1,..M. In the following, a method to identify the regions $R_i^{(j)+}$ is proposed.

Let us consider any "partition" P defined by the boundaries among the regions $R_i^{(j)}$. Each partition is the intersection of K decision regions, where each region is related to one classifier. If the Bayesian boundaries can be expressed as a "combination" of the boundaries of the MCS, then the boundaries of the Bayesian regions cannot be contained into a partition P. In addition, it is easy to see that, for each classifier C_j, the following proposition is true:

$$\left(P \subseteq R_i^{(j)+}\right) \oplus \left(P \subseteq R_i^{(j)-}\right) \qquad (2)$$

The example in figure 1 may help to understand the above concepts. For a problem with three classes, the boundaries defined by two classifiers are displayed. Bayesian boundaries satisfying our assumption are also displayed. These boundaries identify six partitions. Each partition satisfies the condition in equation (2).

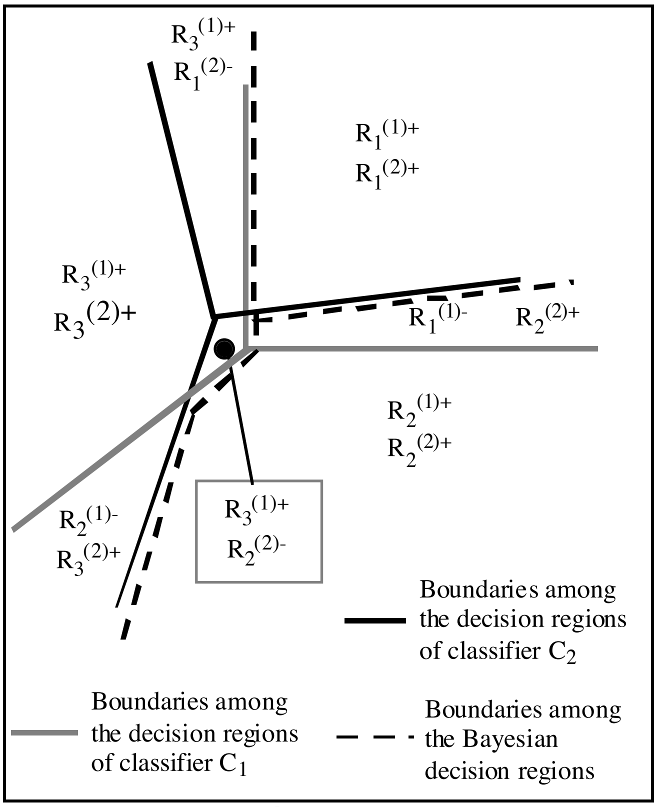

Figure 1

In order to state if $\left(P \subseteq R_i^{(j)+}\right)$ or $\left(P \subseteq R_i^{(j)-}\right)$ for each partition P, the probability of correct classification of each classifier C_j can be computed. The classifier C_j that exhibits the maximum value of the probability of correct classification allows us to state that $P \subseteq R_i^{(j)+}$. Consequently, according to the hypothesis in equation (1), an optimal Bayesian classifier can be obtained by "selecting" the classifier C_j, j = 1,..,K, on the basis of the computation of the probability of correct classification.

In the following, we present two methods for classifier selection based on the estimation of the probability of correct classification in a local region of the feature space surrounding the unknown test pattern. Such local region is computed in terms of the k-nearest neighbors in the training, or validation, data.

3.2 A priori selection method

Let us consider the k-nearest neighbours of the test pattern **X** to be classified. If the k-nearest neighbours forming the "local" region are chosen among the training (or "validation") patterns, for each classifier C_j the probability of classifying correctly the test pattern can be computed as follows:

$$\hat{p}(correct_j) = \frac{N_j}{N} \quad j = 1,..,K \qquad (5)$$

where N_j is the number of "neighbouring" patterns that were correctly classified by the classifier C_j and N is the number of patterns in the neighbourhood. The appropriate size of the neighbourhood should be decided by experiments.

It is worth noticing that the "selection condition" of equation (5) is the same proposed by Woods et al. in [13]. Herein we are pointing out that this selection condition can be used to approximate an "optimal" Bayesian classifier.

If the considered classifiers provide estimates of the class posterior probabilities, the above selection condition can be reformulated as follows.

Given a pattern $\mathbf{X}_i \in \omega_k$ belonging to the neighbourhood, the $\hat{P}^{(j)}(\omega_k \mid \mathbf{X}_i)$ provided by the classifier C_j can be regarded as a measure of the classifier accuracy for the pattern **X**.

Therefore, the selection condition of equation (5) can be rewritten as follows:

$$\hat{p}(correct_j) = \frac{1}{N}\sum_{i=1}^{N} \hat{P}^{(j)}(\omega_k \mid \mathbf{X}_i \in \omega_k) \qquad (6)$$

In order to handle the "uncertainty" in the definition of the neighbourhood size, the posterior probabilities can be "weighted" by the Euclidean distances d_i of the patterns $\mathbf{X}_i$ from **X**:

$$\hat{p}(correct_j) = \frac{\sum_{i=1}^{N} \hat{P}^{(j)}(\omega_k \mid \mathbf{X}_i \in \omega_k) \cdot W_i}{\sum_{i=1}^{N} W_i} \qquad (7)$$

where $W_i = 1/d_i$.

Finally, let us explain the rationale behind the name "a priori selection". According to equation (7), the selection is performed without knowing the class assigned by the classifier C_j to the test pattern **X**.

3.3 A posteriori selection method

If the class assigned by the classifier C_j to the test pattern **X** is known ($C_j(\mathbf{X}) = \omega_k$), this information can be exploited by rewriting equation (5) as follows:

$$\hat{p}\left(correct_j \mid C_j(\mathbf{X}) = \omega_k\right) = = \hat{p}\left(\mathbf{X} \in \omega_k \mid C_j(\mathbf{X}) = \omega_k\right) = \frac{N_{kk}}{\sum_{j=1}^{M} N_{jk}} \tag{8}$$

where N_{kk} is the number of neighbourhood patterns that have been correctly assigned by C_j to the class ω_k, and $\sum_{j=1}^{M} N_{jk}$ is the total number of neighbourhood patterns that have been assigned to the class ω_k by C_j. Equation (8) gives the fraction of neighbourhood patterns assigned to ω_k by C_j that have been correctly classified. This "selection condition" has been also proposed by Woods et al. in [13].

It is easy to see the rationale behind the name "a posteriori" selection. According to equation (8), the selection is performed after knowing the class assigned by the classifier C_j to the test pattern **X**.

If the considered classifiers provide estimates of the class posterior probabilities, the a posteriori selection condition can be formulated as explained in the following. According to the Bayes theorem:

$$\hat{P}\left(\mathbf{X} \in \omega_k \mid C(\mathbf{X}) = \omega_k\right) = = \frac{\hat{P}\left(C(\mathbf{X}) = \omega_k \mid \mathbf{X} \in \omega_k\right)\hat{P}\left(\omega_k\right)}{\sum_{i=1}^{M} \hat{P}\left(C(\mathbf{X}) = \omega_k \mid \mathbf{X} \in \omega_i\right)\hat{P}\left(\omega_i\right)} \tag{9}$$

where $\hat{P}\left(C(\mathbf{X}) = \omega_k \mid \mathbf{X} \in \omega_k\right)$ is the probability to correctly classify the patterns belonging to the class ω_k and it can be estimated, analogously to equation (6), as follows:

$$\hat{P}\left(C(\mathbf{X}) = \omega_k \mid \mathbf{X} \in \omega_k\right) = \frac{1}{\sum_{j=1}^{M} N_{kj}} \sum_{x \in \omega_k} \hat{P}(\omega_k \mid \mathbf{X}) \tag{10}$$

The prior probabilities can be estimated as follows:

$$\hat{P}(\omega_k) = \frac{\sum_{j=1}^{M} N_{kj}}{N} \tag{11}$$

Analogously to the a priori selection, in order to handle the "uncertainty" in the definition of the neighbourhood size, the posterior probabilities can be "weighted" by the Euclidean distances d_i of patterns $\mathbf{X}_i$ from **X**.

If we substitute equations (10) and (11) into equation (9) and introduce the weights $W_i = 1/d_i$, the selection condition of equation (8) can be rewritten as follows:

$$\hat{p}(correct_j \mid C_j(\mathbf{X}) = \omega_k) = \frac{\sum_{\mathbf{X}_i \in \omega_k} \hat{P}_j(\omega_k \mid \mathbf{X}_i) \cdot W}{\sum_{i=1}^{N} \hat{P}_j(\omega_k \mid \mathbf{X}_i) \cdot W} \tag{12}$$

3.4 An algorithm for dynamic classifier selection

In the following, a dynamic classifier selection algorithm based on the two above selection methods is described. Thus, the $\hat{p}(correct_j)$ reported in the following algorithm can be computed according to the "a priori" or "a posteriori" selection methods.

Input parameters: classification results on the training/validation set and size of the neighbourhood

Output: the classification of the test pattern **X**

```
Begin
  For each test pattern X:
  Do If "A posteriori" selection then
  ∀ $C_j$ ($j$ = 1,..,K): compute $C_j(\mathbf{X})$
  If all the classifiers assign X to the same data class $\omega_k$
  then classify X and return $\omega_k$
Else ∀ $C_j$ ($j$ = 1,..,K):
  Begin Do
    STEP 1: Compute $\hat{p}(correct_j)$
    STEP 2: If $\hat{p}(correct_j) < 0.5$ Then Reject $C_j$
  End
  STEP 3: Identify the classifier $C_m$ exhibiting the maximum value of $\hat{p}(correct_j)$, $j = 1,..,K'$, $K' \le K$
  STEP 4: For each $C_l$, $l = 1,..,K'$, compute the following difference $d_l = [\hat{p}(correct_m) - \hat{p}(correct_l)]$
  STEP 5: If $\forall l = 1,..,K'$, $l \ne m$, $d_l >$ Threshold
  Then Select Classifier $C_m$
  Else Select one classifier with $d_l$ < Threshold
End
```

- Steps 1 and 2 focus on selecting *K'* classifiers ($K' \le K$) by removing classifiers that have a probability less than 0.5 to correctly classify the test pattern **X**.
- The differences computed at Step 4 are used to compute a sort of "confidence" for the selection. (If the "a posteriori" selection condition is used then the difference d_j is calculated only for the classifiers that take a decision different from the one taken by the classifier C_m). If all the differences are higher than an a-priori fixed threshold (e.g., 0.1), then there is reasonable confidence that classifier C_m is the most appropriate for the test pattern.
- On the other hand, it is not reasonable to directly select the classifier C_m if there are other classifiers exhibiting similar values of the selecting condition (STEP 5). A random selection among the classifiers exhibiting the higher values of the selecting condition can be

performed. Alternatively, random selection can be substituted by a combination of the classifiers with similar values of the selecting condition, as suggested by Woods et al. [13].

4. Experimental results

The ELENA (Enhanced Learning for Evolutive Neural Architecture) data base was used in our experiments to compare our results to those obtained by Woods et al. [13]. This data base consists of various data sets designed for testing and benchmarking classification algorithms. We used three data sets adopted by Woods et al.: phoneme_CR (French phoneme data), satimage_CR (remote sensing data acquired by the LANDSAT satellite), and texture_CR (data of texture images). Further details on these data sets can be found in [13] or via anonymous ftp at: *ftp.dice.ucl.ac.be* in the directory *pub/neural-nets/ELENA/databases*. In our experiments, we used the same data classes, features, and numbers of training and test samples used by Woods et al.

Our experiments were mainly aimed to:

- compare the performances of selection-based MCSs with the ones of combination-based MCSs based on the most commonly used combination functions (the majority rule and the Bayesian average)
- compare the performances of our selection methods with the ones provided by methods proposed by Woods et al.
- evaluate the "robustness" of our selection methods with respect to the choice of the size of the "neighbourhood" (i.e., the size of the local region used to estimate local accuracies)

First of all, a MCS formed by the same five classification algorithms adopted by Woods et al. was defined (Table1).

Table 1

Classifier	Phoneme	Satimage	Texture
k-nn	87.77%	87.59%	97.75%
MLP	86.29%	84.20%	98.51%
C4.5	84.78%	85.66%	90.95%
QB	75.41%	85.78%	99.04%
LB	73.00%	83.31%	97.42%

Table 1 shows the percentage accuracies of the individual classifiers for the three data sets used. Further details on the design of these classifiers can be found in [13]. According to the work of Woods et al., we randomly partitioned each data set into two equal partitions, keeping the class distributions similar to that of the full data set. Each partition was firstly used as training set and then as test set. In Table 1, accuracies are reported as the average of the two results.

Table 2 shows the performances of various MCSs. All of them are formed by the classifiers of Table 1, but different "operation" mechanisms were used. In particular, the following mechanisms were used:

- The "Overall Class Accuracy" (OCA) selection method proposed by Woods et al.
- The "Local Class Accuracy" (LCA) selection method proposed by Woods et al.
- The A priori and A Posteriori selection methods proposed by the authors (Section 3)

The above experiments were carried out using neighborhood sizes ranging from 1 to 51. For each data set, the results related to the neighbourhood size that provided the highest accuracy have been reported.

Table 2

Classifier	Phoneme	Satimage	Texture
Oracle	97.30	95.85	99.93
Best classifier	87.77	87.59	99.04
OLA	88.53	88.83	99.46
LCA	88.32	90.37	99.44
A priori	88.14	88.23	99.46
A posteriori	88.94	90.81	99,50
Majority rule	89.20	90.88 (3.13)	99.55 (0.9)
Bayesian average	89.71	89.95	99.51

For comparison purposes, the performances of the best individual classifier, the "oracle", and MCSs based on the Majority Rule and the Bayesian Average are also shown (the rejection rate is put within brackets if different from zero). Accuracies provided by combination-based MCSs are sometimes better than the ones of selection-based MCSs. This result is very reasonable, as classifiers very "different", and, therefore, more "independent" were used in these experiments.

With regard to the comparison between our selection methods and the ones proposed by Woods et al., Table 2 shows that the accuracies of our methods are close or better than the ones of Woods et al. In particular, the accuracy of our A Priori Selection is much better than the accuracy provided by the Overall Class Accuracy method. This result points out the advantage of a selection function based on posterior probabilities.

Table 3, shows the results related to the evaluation of the "robustness" of our selection methods with respect to the choice of the size of the "neighbourhood", as compared with the robustness of methods proposed by Woods et al.. To this end, the average accuracy and the standard deviation of the accuracy were computed for values of the neighbourhood sizes ranging from one to fifty one. Our methods provided average accuracies and standard deviations that pointed out their robustness.

Table 3

Classification Algorithm	Phoneme Data Set		Satimage Data Set		Texture Data Set	
	%average accuracy	%standard deviation	%average accuracy	%standard deviation	%average accuracy	%standard deviation
Overall Local Accuracy	87.43	0.69	87.15	0.70	99.21	0.11
A priori Selection	87.42	0.41	86.94	0.41	99.14	0.14
Local Class Accuracy	87.55	0.56	88.90	0.68	99.30	0.10
A posteriori Selection	88.54	0.20	89.60	0.46	99.35	0.11

5. Conclusions

In this paper, we have addressed the "open" research topic of selection-based MCSs. With respect to our previous work reported in [1], a theoretical framework for dynamic classifier selection and another "selection condition" (namely, the a posteriori condition) have been presented. The proposed framework pointed out that dynamic classifier selection can be a method for designing optimal Bayesian classifiers. We think that this "perspective" can stimulate and focus better the future research on selection-based MCSs.

To the best of our knowledge, the work of Woods et al, together with the work of Srihari, represent the "leading edge" for selection-based MCSs. Reported results showed that selection-based MCSs can outperform the combination-based ones. In addition, our selectors exhibited performances that are reasonably close to those of an "oracle". Reported results showed that our selection methods are robust to the variations of the size of the neighbourhood. This is an important achievement, as the choice of the neighbourhood size is a critical point for any method based on local accuracy estimates.

Acknowledgements

The authors wish to thank K.Woods, W.P. Kegelmeyer, and K.Bowyer which provided them with detailed information on the data set used in [13].

References

[1] G. Giacinto and F. Roli, "Adaptive Selection Of Image Classifiers", ICIAP '97, 9th ICIAP, Florence, Italy, Sept 17 - 19, 1997, Lecture Notes in Computer Science 1310, Springer Verlag Ed., pp.38-45

[2] L. Xu, A. Krzyzak, and C.Y. Suen, "Methods for combining multiple classifiers and their applications to handwriting recognition", IEEE Trans. on Systems, Man, and Cyb., Vol. 22, No. 3, May/June 1992, pp. 418-435

[3] Y.S. Huang, and C.Y. Suen, "A method of combining multiple experts for the recognition of unconstrained handwritten numerals", IEEE Trans. on PAMI, Vol.17, No.1, January 1995, pp.90-94

[4] S. N. Srihari et al., "Decision combination in multiple classifiers systems", IEEE Trans. on PAMI, Vol.16, No.1, Jan. 1994, pp. 66-75

[5] J. Kittler et al., "On Combining Classifiers", IEEE Trans. on PAMI, Vol.20, No.3, March 1998, pp. 226-239

[6] L. K. Hansen, and P. Salamon, "Neural network ensembles", IEEE Trans. on PAMI, Vol. 12, No. 10, October 1990, pp. 993-1001

[7] A. J. C. Sharkey (Ed.), Special Issue: Combining Artificial Neural Nets: Ensemble Approaches. Connection Science Vol. 8, No. 3 & 4, Dec. 1996

[8] K. Tumer and J. Gosh, "Error correlation and error reduction in ensemble classifiers", Connection Science 8, December 1996, pp. 385-404

[9] B. E. Rosen, "Ensemble learning using decorrelated neural networks", Connection Science 8, December 1996, pp. 373-383

[10] D. Partridge, W. B. Yates, "Engineering multiversion neural-net systems", Neural Computation, 8, 1996, pp. 869-893

[11] C. Y. Suen et al., "The combination of multiple classifiers by a neural network approach", Int. Journal of Pattern Recognition and Artificial Intelligence, Vol. 9, no.3, 1995, pp.579-597

[12] G. Giacinto and F. Roli, "Ensembles of Neural Networks for Soft Classification of Remote Sensing Images", Proc. of the European Symposium on Intelligent Techniques, Bari, Italy, pp. 166-170

[13] K. Woods et al., "Combination of multiple classifiers using local accuracy estimates", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.19, No.4, April 1997, pp. 405-410

[14] A. J. C. Sharkey (Ed.), Special Issue: Combining Artificial Neural Nets: Modular Approaches. Connection Science 9, No.1, March 1997