In Proc. NAFIPS'98, Pensacola, Florida, 1998, 193-197

On Combining Multiple Classifiers by Fuzzy Templates

Ludmila I. Kuncheva
School of Mathematics, University of Wales, Bangor
Bangor, Gwynedd LL57 1UT, United Kingdom
e-mail: L.I.Kuncheva@bangor.ac.uk

James C. Bezdek* and Melanie A. Sutton
Department of Computer Science, University of West Florida
Pensacola, FL 32514, USA
e-mail: jbezdek@argo.cs.uwf.edu and msutton@uwf.edu

Abstract

We study classifier fusion by the fuzzy template (FT) technique. Given an object to be classified, each classifier from the pool yields a vector with degrees of "support" for the classes, thereby forming a decision profile. A fuzzy template is associated with each class as the averaged decision profile over the training samples from this class. A new object is then labeled with the class whose fuzzy template is closest to the objects' decision profile. We give a brief overview of the field to place the FT approach in a proper group of classifier combination techniques. Experiments with two data sets (satimage and phoneme) from the ELENA database demonstrate the superior performance of FT over: a version of majority voting; aggregation by fuzzy connectives (minimum, maximum, and product); and (unweighted) average.

1 Introduction

We consider a pattern classification problem where $x \in \Re^p$ is a feature vector to be labeled into one or more of c classes. Every mapping $D : \Re^p \to \{1, 2, \ldots, c\}$ is called a *crisp classifier*. A *soft classifier* is a mapping

$$\tilde{D} : \Re^p \to [0,1]^c - \{\mathbf{0}\},$$

$$\mu_{\tilde{D}}(x) = [\mu^1_{\tilde{D}}(x), \ldots, \mu^c_{\tilde{D}}(x)]^T, \;\; x \in \Re^p. \tag{1}$$

The decision of $\tilde{D}$ can be "hardened" so that a crisp class label from $1, 2, \ldots, c$ is assigned to x.

Many classifiers produce class labels as posterior probabilities for the classes, given x, i.e. $\mu^i_{\tilde{D}}(x) = P(i|x), i = 1, \ldots, c$. Since we do not restrict $\mu^i_{\tilde{D}}(x), i = 1, \ldots, c$ to add up to one, our model allows the labels to be interpreted as possibilistic, probabilistic or fuzzy.

Combining multiple classifiers to achieve higher accuracy is called different names in the literature:

- combination of multiple classifiers [22, 26, 30, 31];
- classifier fusion [5, 10, 16];
- mixture of experts [13, 14, 15, 25];
- committees of neural networks [3, 7];
- consensus aggregation [2, 24];
- voting pool of classifiers [1];
- dynamic classifier selection [30];
- composite classifier system [6];
- classifier ensembles [7, 9].

Other catchy names include: divide-and-conquer classifiers [4]; pandemonium system of reflective agents [27]; change-glasses approach to classifier selection [17], etc. The paradigms differ on the assumptions about the classifiers (e.g., dependencies and output type), on the aggregation strategy (e.g., global (classifier *fusion*) or

*Research supported by ONR grant N00014-96-1-0642.

Output	No re-training	Re-training
Crisp	*Majority* [23]	Knowledge-Behavior Space [12]
Fuzzy	Fuzzy connectives (e.g., *Min, Max,* OWA) [19], *Average, Product,* [29] "Naive" Bayes combination [31]	Trained linear combination [11, 28], Neural network [15], Fuzzy integral [5, 16], Dempster-Shaffer combination [26], **Fuzzy templates** [20]

Table 1: Classifier fusion techniques

local (classifier *selection*)), and the aggregation procedure (e.g., a function, a neural network, or an algorithm), etc. The great majority of aggregation techniques are heuristic and empirical. Majority voting has been investigated in more detail [23].

When designing a combination scheme we can choose either to train or not to train at the aggregation or second level. Many aggregation rules do not use any further training after the classifiers have been designed. Typically these are simple aggregation rules like majority voting, various forms of averaging, or fuzzy set connectives [18, 19, 21]. The reason for not training at the second level is that for small data sets this can easily lead to overtraining (overfitting the data).

Table 1 systematizes some classifier *fusion* paradigms with respect to 2 factors: • classifier output – crisp (single label) or soft (a vector $\mu_{\tilde{D}}(x) \in [0,1]^c - \{\mathbf{0}\}$); and • re-training (using training at the second level). Marked in boldface in Table 1 is the **fuzzy template** (FT) technique, and in italic are the competing classifier fusion techniques that we study here.

Amongst the techniques in the bottom right cell of Table 1, FT is the simplest one. We chose simple techniques for comparison because it is always advisable to try the simple methods first [24] and *if they fail*, more sophisticated methods should be tried. An example of a simple, accurate and robust technique is majority voting [1, 22].

In Section 2 we present the fuzzy template technique for combining multiple classifiers. Section 3 contains our experiments with 2 data sets (**Satimage** and **Phoneme** from the ELENA database), and Section 4 provides conclusions.

2 Fuzzy templates for combining multiple classifiers

Let $C = \{C_1, \ldots, C_L\}$ be the set of L classifiers. We denote by $C_i(x) = [d_{i,1}(x), \ldots, d_{i,c}(x)]^T$ the output of the ith classifier, $x \in \Re^p$, where $d_{i,j}(x)$ is the degree of "support" given by classifier C_i to the hypothesis that x comes from class j. Typically, $d_{i,j}(x) \in [0,1]$. Besides the usual interpretation as an estimate of the posterior probability $P(j|x)$ by classifier C_i, $d_{i,j}(x)$ can be viewed as typicalness, belief, certainty, possibility, etc., not necessarily coming from statistical classifiers. $\tilde{D}$ is then

$$\tilde{D}(x) = \mathcal{F}(C_1(x), \ldots, C_L(x)) \tag{2}$$

where $\mathcal{F}$ is called the *aggregation rule.*

Definition 1. *The decision profile of a combination of L classifiers given $x \in \Re^p$ is the matrix*

$$DP(x) = \begin{bmatrix} d_{1,1}(x) & \ldots & d_{1,j}(x) & \ldots & d_{1,c}(x) \\ \ldots & & & & \\ d_{i,1}(x) & \ldots & d_{i,j}(x) & \ldots & d_{i,c}(x) \\ \ldots & & & & \\ d_{L,1}(x) & \ldots & d_{L,j}(x) & \ldots & d_{L,c}(x) \end{bmatrix}.$$

Definition 2. *Let $Z = \{Z_1, \ldots, Z_N\}$, $Z_j \in \Re^p$ be crisply labeled training data. The fuzzy template of class i*

is the $L \times c$ matrix $F_i = \{f_i(k,s)\}$ whose elements are obtained by [20]:

$$f_i(k,s) = \frac{\sum_{j=1}^{N} \; Ind(Z_j,i) \; d_{k,s}(Z_j)}{\sum_{j=1}^{N} \; Ind(Z_j,i)} \tag{3}$$

where $Ind(Z_j,i)$ is an indicator function with value 1 if Z_j has label i, and 0 otherwise.

Thus, the fuzzy template for class i is the average of the decision profiles of the elements of the training set Z labeled in class i.

When $x \in \Re^p$ is submitted for classification, the FT combination of classifiers produces the soft class label vector [20] with components:

$$\mu^i_{\tilde{D}}(x) = \mathcal{S}(F_i, DP(x)), \tag{4}$$

where $\mathcal{S}$ is interpreted as *similarity.* Regarding the two arguments as fuzzy sets, various fuzzy measures of similarity can be used. Here we used

$$\mu^i_{\tilde{D}}(x) = 1 - \frac{1}{L\;c} \sum_{k=1}^{L} \sum_{s=1}^{c} \; (f_i(k,s) - d_{k,s}(x))^2. \tag{5}$$

To "harden" the classification decision, x is assigned to the class label with the maximal support. For comparison we also used the following L-place operators as the aggregation rule ($\mathcal{F}$): majority vote, minimum, maximum, average, and product, i.e.,

$$\mu^i_{\tilde{D}}(x) = \mathcal{F} \;\left(d_{1,i}(x), \ldots, d_{L,i}(x) \right), \;\; i = 1, \ldots, c. \tag{6}$$

3 Experiments

We considered two data sets from the ELENA database available via anonymous ftp at
`ftp.dice.ucl.ac.be`,
directory
`pub/neural-nets/ELENA/databases`.
Results with the same data using other combination methods can be found in [30].

The **Satimage** data was generated from the Landsat Multi-Spectral Scanner image data. It consists of 6435 patterns (pixels) with 36 attributes (4 spectral bands × 9 pixels in a 3x3 neighborhood). Pixels are crisply labeled in one of 6 classes, and are presented in random order in the database. The classes are: red soil (23.82 %), cotton crop (10.92 %), grey soil (21.10 %), damp grey soil (9.73 %), soil with vegetation stubble (10.99 %), and very damp grey soil (23.43 %). What makes this database attractive is: large sample size; numerical, equally ranged features; no missing values; compact classes of approximately equal size, shape and prior probabilities. Figure 1 is a scatterplot of the 6 `satimage` classes on features # 17 and # 18. In our experiments we used only features # 17 to # 20, as recommended by the database designers.

The **Phoneme** data consists of 5404 5-dimensional vectors characterizing two classes of phonemes: nasals (70.65 %) and orals (29.35 %). The scatterplot on features 3 and 4 of 800 randomly selected data points is shown in Figure 2. The two classes are highly overlapping with complex classification boundaries suggesting that parametric classifiers will not achieve good results.

We designed 6 classifiers with the **Satimage** data and 10 classifiers with the **Phoneme** data using all combinations of 2 features in each case. Three sets of such classifiers were considered: Linear Discriminant Classifier (LDA); Quadratic Discriminant Classifier (QDA); and Logistic Classifier (LOG). We used the Matlab code from the package PRTOOLS [8].

With each data set we made 4 random splits into training and test sets with 200 elements for training and the rest 6235 (**Satimage**) or 5204 (**Phoneme**) for testing. Tables 2 and 3 show the *test* classification accuracy averaged over the 4 splits. We display also the averaged test accuracy of the single best classifiers.

We did not consider here the rejection option – all ties were broken randomly. Therefore in our version of majority voting we assigned a class label, even if the number of votes for the majority class was less than half (for the **Satimage** data where 6 classes are possible, the votes may be spread so that none of the classes gets more than half of all votes).

In all experiments but one with the **Satimage** data the combinations achieved better accuracy than the best single classifiers. Amongst these, FT produced the highest improvement. As expected, the **Phoneme**

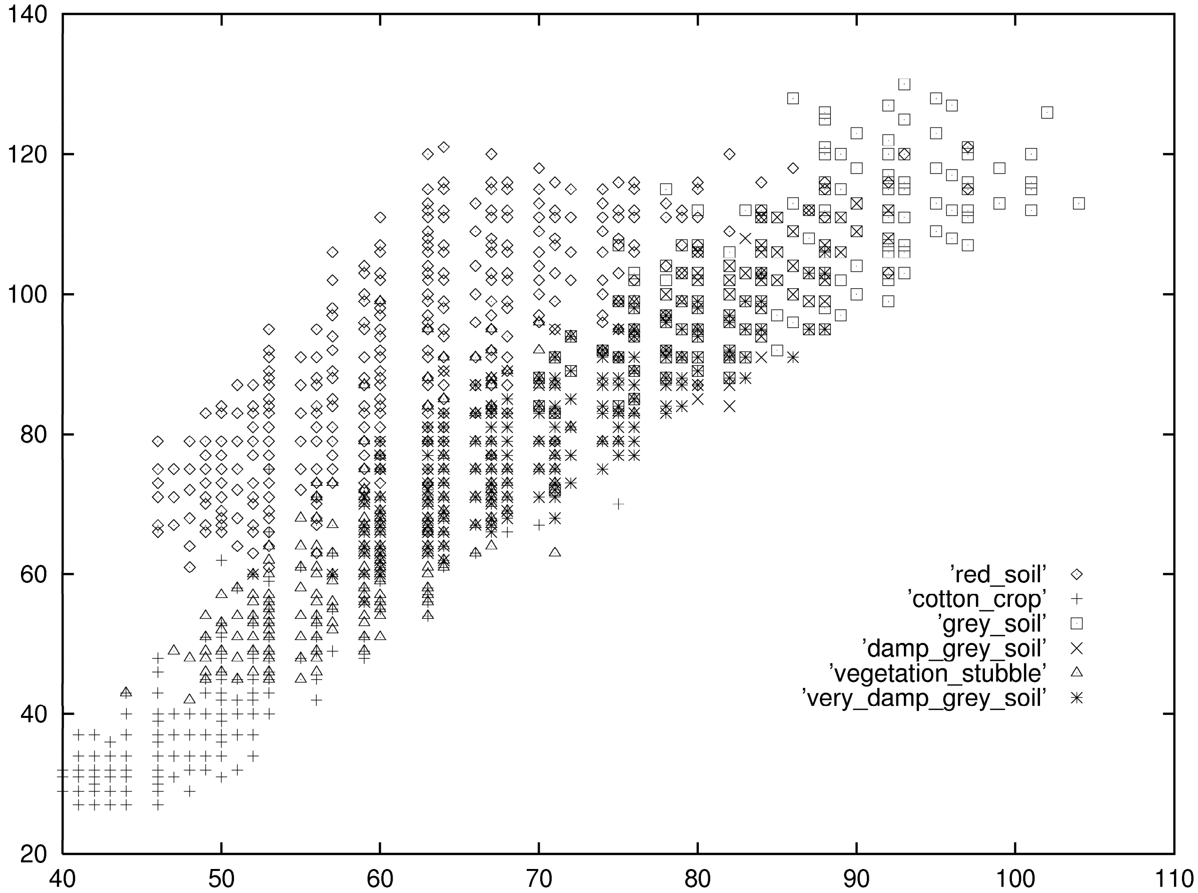

Figure 1: Scatterplot of the Satimage data on features # 17 and # 18

data appeared to be difficult for the two parametric classifiers (LDA, QDA) and the semi-parametric classifier (LOG). In this case the fuzzy template technique clearly outperformed the rest since it was the only one that consistently improved on the single classifier rate.

Majority voting could have produced better results if we had used the suggestion in [23] to double the best classifier's vote or to drop the worst classifiers' vote in order to get an odd number of voters.

Unlike the rest of the aggregation techniques selected for comparison here, FT uses the data set to (re)train the classifier combination. This probably explains the higher accuracy.

4 Conclusions

In this paper we discussed *fuzzy templates* for combining multiple classifiers. In our experiments on two data sets this scheme was superior to majority vote, maximum, minimum, average and product aggregation connectives.

In our previous study [20] we experimented with "distorting" one of the classifiers in the pool by substituting $\sqrt{\mu_{\tilde{D}}}$ for $\mu_{\tilde{D}}$. The fuzzy templates still outperformed the maximum, minimum, and average aggregation rules used in this study.

It is interesting to compare the fuzzy templates with other combination techniques from the same group (Table 1). We expect that FT will generalize better than techniques that use large numbers of parameters that are estimated during re-training, e.g., fuzzy integrals. This is important when the training data set is not large (like in this study) so that re-using it may lead to an unacceptable overfit.

Euclidean distance is not the only option for calculating the similarity as in Eqn. (5). We can use various

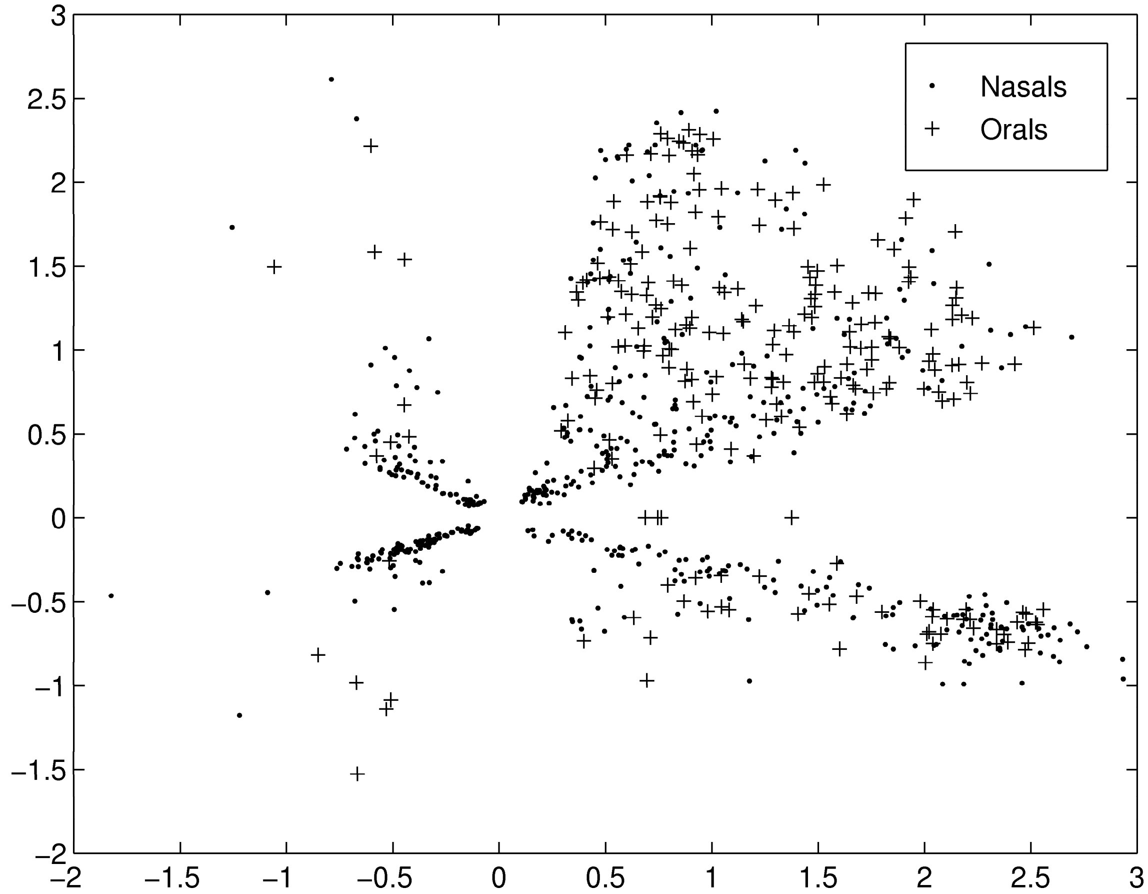

Figure 2: Scatterplot of the Phoneme data on features # 3 and # 4

other measures from fuzzy set theory. This will be the topic of a future study.

An option that may improve the overall accuracy is to *select* a subset from the pool of classifiers instead of using all of them. This will put all the combination rules in the re-training group. The method of selection can vary from exhaustive search (in the case of a few classifiers) to procedures borrowed from feature selection or from editing methods for the k-nearest neighbor rule.

In summary, we feel that *FT* warrants further study because it is simple, achieves high accuracy, and generalizes well.

References

[1] R. Battiti and A.M. Colla. Democracy in neural nets: Voting schemes for classification. *Neural Networks*, 7:691–707, 1994.

[2] J.A. Benediktsson and P.H. Swain. Consenus theoretic classification methods. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:688–704, 1992.

[3] C.M. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[4] C.-C. Chiang and H.-C. Fu. A divide-and-conquer methodology for modular supervised neural network design. In *IEEE International Conference on Neural Networks*, pages 119–124, Orlando, Florida, 1994.

[5] K.B. Cho and B.H. Wang. Radial basis function based adaptive fuzzy systems. In *Proc. FUZZ/IEEE'95*, pages 247–252, Yokohama, Japan, 1995.

Classifier design	Single best	Majority (ties [%])	Maximum	Minimum	Average	Product	**Fuzzy templates**
LDA	78.54	73.72 (13.10)	80.42	81.45	80.14	80.72	**81.72**
QDA	79.88	81.71 (5.38)	82.62	83.99	83.59	84.18	**84.48**
LOG	74.20	75.02 (5.93)	79.21	79.78	78.60	79.17	**81.93**

Table 2: Averaged test classification accuracy over 4 random splits of 200/6235 with Satimage data

Classifier design	Single best	Majority (ties [%])	Maximum	Minimum	Average	Product	**Fuzzy templates**
LDA	73.69	72.52 (4.98)	72.95	72.95	72.78	72.80	**75.35**
QDA	75.86	76.66 (5.44)	75.65	75.65	76.13	76.11	**76.27**
LOG	73.30	72.27 (4.49)	72.98	72.98	72.58	72.61	**74.03**

Table 3: Averaged test classification accuracy over 4 random splits of 200/5204 with Phoneme data

[6] B.V. Dasarathy and B.V. Sheela. A composite classifier system design: concepts and methodology. *Proceedings of IEEE*, 67:708–713, 1978.

[7] H. Drucker, C. Cortes, L.D. Jackel, Y. LeCun, and V. Vapnik. Boosting and other ensemble methods. *Neural Computation*, 6:1289–1301, 1994.

[8] R.P.W. Duin. PRTOOLS(Version 2). A Matlab toolbox for pattern recognition. Pattern Recognition Group, Delft University of Technology, June 1997.

[9] E. Filippi, M. Costa, and E. Pasero. Multy-layer perceptron ensembles for increased performance and fault-tolerance in pattern recognition tasks. In *IEEE International Conference on Neural Networks*, pages 2901–2906, Orlando, Florida, 1994.

[10] M. Grabisch and F. Dispot. A comparison of some for fuzzy classification on real data. In *2nd International Conference on Fuzzy Logic and Neural Networks*, pages 659–662, Iizuka, Japan, 1992.

[11] S. Hashem, B. Schmeiser, and Y. Yih. Optimal linear combinations of neural networks: an overview. In *IEEE International Conference on Neural Networks*, pages 1507–1512, Orlando, Florida, 1994.

[12] Y.S. Huang and C.Y. Suen. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:90–93, 1995.

[13] R.A. Jacobs. Methods for combining experts' probability assessments. *Neural Computation*, 7:867–888, 1995.

[14] R.A. Jacobs, M.I. Jordan, S.J. Nowlan, and G.E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87, 1991.

[15] M.I. Jordan and L. Xu. Convergence results for the EM approach to mixtures of experts architectures. *Neural Networks*, 8:1409–1431, 1995.

[16] J.M. Keller, P. Gader, H. Tahani, J.-H. Chiang, and M. Mohamed. Advances in fuzzy integration for pattern recognition. *Fuzzy Sets and Systems*, 65:273–283, 1994.

[17] L.I. Kuncheva. Change-glasses approach in pattern recognition. *Pattern Recognition Letters*, 14:619–623, 1993.

[18] L.I. Kuncheva. A fuzzy aggregation of multiple classification decisions. *Control and Cybernetics*, 25:337–352, 1996.

[19] L.I. Kuncheva. An application of OWA operators to the aggregation of multiple classification decisions. In R.R. Yager and J. Kacprzyk, editors, *The Ordered Weighted Averaging operators. Theory and Applications*, pages 330–343. Kluwer Academic Publishers, USA, 1997.

[20] L.I. Kuncheva, R.K. Kounchev, and R.Z. Zlatev. Aggregation of multiple classification decisions by fuzzy templates. In *Third European Congress on Intelligent Technologies and Soft Computing EUFIT'95*, pages 1470–1474, Aachen, Germany, August 1995.

[21] L.I. Kuncheva and R. Krishnapuram. A fuzzy consensus aggregation operator. *Fuzzy Sets and Systems*, 79:347–356, 1996.

[22] L. Lam and C.Y. Suen. Optimal combination of pattern classifiers. *Pattern Recognition Letters*, 16:945–954, 1995.

[23] L. Lam and C.Y. Suen. Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics*, 27(5):553–568, 1997.

[24] K.-C. Ng and B. Abramson. Consensus diagnosis: A simulation study. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:916–928, 1992.

[25] S.J. Nowlan and G.E. Hinton. Evaluation of adaptive mixtures of competing experts. In R.P. Lippmann, J.E. Moody, and D.S. Touretzky, editors, *Advances in Neural Infprmation Processing Systems 3*, pages 774–780, 1991.

[26] G. Rogova. Combining the results of several neural network classifiers. *Neural Networks*, 7:777–781, 1994.

[27] F. Smieja. The pandemonium system of reflective agents. *IEEE Transactions on Neural Networks*, 7:97–106, 1996.

[28] V. Tresp and M. Taniguchi. Combining estimators using non-constant weighting functions. In G. Tesauro, D.S. Touretzky, and T.K. Leen, editors, *Advances in Neural Information Processing Systems 7*, Cambridge, MA, 1995. MIT Press.

[29] M. van Breukelen, R.P.W Duin, D.M.J. Tax, and J.E. den Hartog. Combining classifiers for the recognition of handwritten digits. In *I-st IAPR TC1 Workshop on Statistical Techniques in Pattern Recognition*, pages 13–18, Prague, Czech Republic, 1997.

[30] K. Woods, W.P. Kegelmeyer, and K. Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:405–410, 1997.

[31] L. Xu, A. Krzyzak, and C.Y. Suen. Methods of combining multiple classifiers and their application to handwriting recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 22:418–435, 1992.