Hierarchical Neural Networks Utilising Dempster-Shafer Evidence Theory

Rebecca Fay, Friedhelm Schwenker, Christian Thiel, and Günther Palm

University of Ulm
Department of Neural Information Processing
D-89069 Ulm, Germany
{rebecca.fay, friedhelm.schwenker, christian.thiel@uni-ulm.de,
guenther.palm}@uni-ulm.de

Abstract. Hierarchical neural networks show many benefits when employed for classification problems even when only simple methods analogous to decision trees are used to retrieve the classification result. More complex ways of evaluating the hierarchy output that take into account the complete information the hierarchy provides yield improved classification results. Due to the hierarchical output space decomposition that is inherent to hierarchical neural networks the usage of Dempster-Shafer evidence theory suggests itself as it allows for the representation of evidence at different levels of abstraction. Moreover, it provides the possibility to differentiate between uncertainty and ignorance. The proposed approach has been evaluated using three different data sets and showed consistently improved classification results compared to the simple decision-tree-like retrieval method.

1 Introduction

Hierarchical neural networks have proven suitable for pattern recognition and show many benefits when applied to classification problems of various kind [1][2][3][4][5]. Simple evaluation strategies like retrieving the accumulated classification result in a decision-tree-like manner yield good classification results. Despite all the advantages this simple method features, such as rather short classification time and availability of intermediate results, a major disadvantage is the missing ability to correct misclassifications that occur at higher levels of the hierarchy. Hence it would be beneficial not only to take a single path within the hierarchy into account but to consider all classifiers of the hierarchy.

Due to the inherent characteristics of the hierarchy there are several constraints such a comprehensive evaluation approach should meet. The classifier hierarchy naturally provides a hierarchical class grouping, i.e. the individual classifiers provide results for not necessarily single classes but sets of classes. Thus the evaluation method should provide means of dealing with information provided at different levels of abstraction without enforcing to assign information at a more detailed level than is justified. Moreover, attributed to the fact, that not all classifiers within the hierarchy provide information about all classes, but only

F. Schwenker and S. Marinai (Eds.): ANNPR 2006, LNAI 4087, pp. 198–209, 2006.

deal with a specific subset of classes, there must be a possibility to state that a given sample belongs to an unknown class. Therefore it is necessary that the eligible approach offers a possibility to represent lack of knowledge and doubt.

Taking this into consideration the Dempster-Shafer evidence theory seems to be applicable as it fulfills the above mentioned constraints.

2 Method

In the following hierarchical neural networks are introduced. This includes the generation of the classifier hierarchies as well as the training of the hierarchy. Furthermore, two methods for hierarchy evaluation are presented: a simple decision-tree-like method and a more complex method based on Dempster-Shafer evidence theory. The proposed evidence theoretic approach only concerns the hierarchy evaluation. The hierarchy generation and training is the same for both methods.

2.1 Dempster-Shafer Evidence Theory

Dempster-Shafer evidence theory [6][7] is a mathematical theory of evidence and plausibility reasoning. It provides means of representing and combining measures of evidence. Major advantages of this theory are the ability to discriminate between ignorance and uncertainty, the ability to easily represent evidence at different levels of abstraction and the possibility to combine evidence from different sources. In the following the basic concepts of the Dempster-Shafer evidence theory are briefly explained.

Let Ω be a finite set of q mutually exclusive atomic hypotheses $\Omega = \{\theta_1, ..., \theta_q\}$ called the *frame of discernment* representing the universe of discourse and let 2^Ω denote the power set of Ω.

A *basic probability assignment* or *mass function* m over a frame of discernment Ω is a function $m : 2^\Omega \mapsto [0, 1]$ that satisfies the following two conditions:

$$m(\emptyset) = 0 \text{ and } \sum_{A \subseteq \Omega} m(A) = 1 \tag{1}$$

The mass $m(A)$ specifies the belief in hypothesis A which does not need to be atomic, but can be a set of atomic hypothesis. In that case $m(A)$ reflects ignorance as it is not possible to further subdivide the belief in A among the subsets of A. Thus the mass $m(A)$ specifies the degree of belief that is assigned to exactly the set $A \subseteq \Omega$ and not to any subset of A.

With m being a basic probability assignment the *belief function* $Bel : 2^\Omega \mapsto [0, 1]$ is defined as follows:

$$Bel(A) = \sum_{B:B \subseteq A} m(B) \tag{2}$$

If m is a basic probability assignment the *plausibility function* $Pl : 2^\Omega \mapsto [0, 1]$ is defined as:

$$Pl(A) = \sum_{B:A \cap B \neq \emptyset} m(B) \tag{3}$$

Two basic probability assignments m_1 and m_2 from two independent sources can be combined via Dempster's combination rule, the so called *orthogonal sum* $m_{1,2} = m_1 \oplus m_2$ which is defined as:

$$m_{1,2}(C) = K^{-1} \sum_{A,B:A\cap B=C} m_1(A) \cdot m_2(B), \forall C \neq \emptyset \tag{4}$$

where K is a measure for the conflict between the two sources. The conflict K is defined as:

$$K = 1 - \sum_{A,B:A\cap B=\emptyset} m_1(A) \cdot m_2(B) = \sum_{A,B:A\cap B\neq\emptyset} m_1(A) \cdot m_2(B) \tag{5}$$

The orthogonal sum $m_1 \oplus m_2$ does only exists if $K \neq 0$ and the result $m_{1,2}$ is then a basic probability assignment. Otherwise the two sources are said to be totally contradictory.

Within the transferable belief model [8] positive masses can be assigned to the empty set $\emptyset$ entailing unnormalised belief functions [9]:

$$m_{1,2}(C) = \sum_{A,B:A\cap B=C} m_1(A) \cdot m_2(B), \forall C \subseteq \Omega \tag{6}$$

A high value for the mass of the empty set $\emptyset$ indicates a high conflict between the sources.

2.2 Hierarchical Neural Networks

Hierarchial neural networks consist of serval simple neural networks that are hierarchically organised. Thus the nodes within the hierarchy represent individual neural classifiers.

The basic idea of hierarchical neural networks is the hierarchical decomposition of a complex classification problem into several less complex ones. This yields hierarchical class groupings splitting the decision process into multiple steps exploiting rough to fine classification. The hierarchy emerges from recursive partitioning of the original set of classes C into several disjoint subsets C_i until the subsets consisting of single classes result. C_i is the subset of classes to be classified by node i, where i is a recursively composed index reflecting the path from the root node to node i. The subset C_i of node i is decomposed into s_i disjoint subsets $C_{i,j}$, where $C_{i,j} \subset C_i$, $C_i = \cup_{j=0}^{s_i-1} C_{i,j}$ and $C_{i,j} \cap C_{i,k} = \emptyset$, $j \neq k$. The total set of classes $C_0 = C$ is assigned to the root node. Consequently nodes at higher levels of the hierarchy classify between larger subsets of classes whereas nodes at the lowest level discriminate between single classes. This divide-and-conquer strategy yields several simple classifiers, that are more easily manageable, instead of one extensive classifier. These simple classifiers can be amended much more easily to the decomposed simple classification tasks than one classifier could be adapted to the original complex classification task. Furthermore different feature types X_i are used within the hierarchy. For each classification task the feature type that allows for the best discrimination is chosen. An example of such a hierarchy is shown in figure 1.

Hierarchy Generation. The hierarchy is generated by unsupervised k-means clustering. In order to decompose the set of classes C_i assigned to one node i into s_i disjoint subsets a k-means clustering is performed with all data points $\{x^\mu \in X_i | t^\mu \in C_i\}$ belonging to these classes. Depending on the distribution of the classes across the k-means clusters s_i disjoint subsets $C_{i,j}$ are formed. One successor node j corresponds to each subset. For each successor node j again a k-means clustering is performed to further decompose the corresponding subset $C_{i,j}$. The k-means clustering is performed for each feature type. The different clusterings are evaluated and the clusterings which group data according to their class labels are preferred. Since the k-means algorithm depends on the initialisation of the clusters, k-means clustering is performed several times per feature type. In this study the number of k-means clustering runs per feature type was 10.

The number of clusters k must be at least the number of successor nodes or the number of subsets s respectively but can also exceed this number. If the number of clusters is higher than the number of successor nodes, several clusters are grouped together so that the number of groups equals the number of successor nodes. All possible groupings are evaluated. In the following all equations only refer to clusterings for reasons of simplicity, i.e. the number of clusters k equals the number of successor nodes s. A valuation function is used to rate the clusterings or groupings respectively. The valuation function prefers clusterings that group data according to their class labels. Clusterings where data is uniformly distributed across clusters notwithstanding their class labels receive low ratings. Furthermore clusterings are preferred which evenly divide the classes. Thus the valuation function rewards unambiguity regarding the class affiliation of the data assigned to a prototype as well as uniform distribution regarding the number of data points assigned to each prototype.

The valuation function $V(p)$ consists of two terms regulated by a scaling parameter $\lambda > 0$. The first term $E(p)$ calculates the entropy of the distribution of each class across the different clusters. This accounts for unambiguous distribution of the data considering the corresponding classes. The term $E(p)$ becomes minimal if it is ensured for all classes that all data belonging to one class is indeed assigned to one cluster. It becomes maximal if all data belonging to one class is uniformly distributed across all clusters. The second term $D(p)$ computes the deviation from the uniform distribution. This term becomes minimal if each cluster is assigned the same number of data points. This allows for the even division of the classes into subsets. During the hierarchy generation phase we are looking for clusterings that minimise the valuation function $V(p)$. The influence of the respective term is regulated by the scaling parameter λ. Both terms are normalised so that they return values in the interval $[0, 1]$. The valuation function $V(p)$ is given by

$$V(p) = \frac{1}{l \log_2(k)} E(p) + \lambda \frac{1}{l(k-1)} D(p) \to \min \tag{7}$$

where $E(p) = -\sum_{i=1}^{l} \sum_{j=1}^{k} p_i^j \log_2(p_i^j)$ and $D(p) = \sum_{j=1}^{k} |\sum_{i=1}^{l} p_i^j - \frac{l}{k}|$ with $p_i^j = \frac{|X_i \cap Z_j|}{|X_i|}$ denoting the rate of patterns from class i, that belong to cluster

j. Here $X_i = \{x_\mu | \mu = 1, ..., M; t^\mu = i\} \subseteq X$ is the set of data points that belong to class i, $R_j = \{x \in \mathbb{R}^d | j = \text{argmin}_{i=1,...,k} \|x - z_i\|\}$ denotes the Voronoi cell defined by cluster j and $Z_j = R_j \bigcap X$ is the set of data points that were assigned to cluster j. The center of cluster i is z_i. The best clustering, i.e. the one that minimises the valuation function $V(p)$, is chosen and used for determining the division of the set of classes into subsets. Moreover this also determines which feature type will be used for the corresponding classifier. So each classifier within the hierarchy can potentially use its own feature type. To identify which classes will be added to which subset the distribution of the data across the clusters is considered. The division in subsets C_j is carried out by maximum detection. The set of classes belonging to subset C_j is defined as $C_j = \{i \in C | j = \text{argmax}\{q_{i,1}, ..., q_{i,k}\}\}$ where $q_{i,j} = \frac{|X_i \cap Z_j|}{|Z_j|}$ denotes the rate of class i in cluster j. For each class it is determined to which cluster j^* the majority of data points belonging to this class were associated. The class label will then be added to the corresponding subset C_{j^*}.

To generate the hierarchy at first the set of all classes is assigned to the root node. Starting with a clustering on the complete data set the set of classes is divided into subsets. Each subset is assigned to a successor node of the root node. Now the decomposition of the subsets is continued until no further decomposition is possible or until the decomposition does not lead to a new division. An example of a classification hierarchy is shown in figure 1.

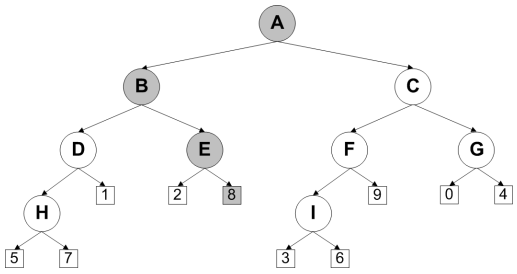

Fig. 1. Classifier hierarchy generated for the classification of 10 classes. Each node within the hierarchy represents a neural network which is used as a classifier. The end nodes represent classes. To each node a feature type and a set of classes are assigned. The corresponding neural network uses the assigned feature type to discriminate between the assigned classes. The highlighted path (in grey) shows the nodes activated during the classification of a sample that is classified as class 8.

Training and Classification. The hierarchy is trained by separately training the individual classifiers with the data $\{x^\mu \in X_i | t^\mu \in C_i\}$ that belong to the subsets of classes assigned to each classifier. For the training the respective feature type X_i identified during the hierarchy generation phase is used. The data will be relabelled so that all data points of the classes belonging to one subset $C_{i,j}$ have the same label j, i.e. $\tilde{t}^\mu = j, x^\mu \in X_i, t^\mu \in C_{i,j}$. The number of input neurons of the single classifiers is defined by the dimension d_i of the

respective feature type X_i assigned to the corresponding node i. The number of output nodes equals the number of successor nodes s_i. The classifiers are trained using supervised learning algorithms. The classifiers within the hierarchy can be trained independently, i.e. all classifiers can be trained in parallel.

Within the hierarchy different types of classifiers can be used. Examples of classifiers would be radial basis function networks, linear vector quantisation classifiers [5] or support vector machines [4]. We chose RBF networks as classifiers. They were trained with a three phase learning algorithm [10].

One way to obtain the classification result is similar to the retrieval process in a decision tree. Starting with the root node the respective feature vector of the object to be classified is presented to the trained classifier. By means of the classification output the next classifier to categorise the data point is determined, i.e. the classifier j^* corresponding to the highest output value $o(j^*)$ is chosen such that $j^* = argmax_{j=1..s_i}(o(j))$. Thus a path through the hierarchy from the root node to an end node is obtained which not only represents the class of the object but also the subsets of classes to which the object most likely belongs. This means that the data point is not presented to all classifiers within the hierarchy and the hierarchical decomposition of the classification problem yields additional intermediate information.

If only intermediate results are of interest it is not necessary to evaluate the complete path. In order to solve a task it might be sufficient to know whether the object to be recognised belongs to a set of classes and the knowledge of the specific category of the object might not add any value. If the task for example is to grasp a cup, it is not necessary to distinguish between red and green cups. Moreover, when looking for a specific object it might in some cases not be necessary to retrieve the final classification result if a decision at a higher level of the hierarchy already excludes this object.

2.3 Utilising Dempster-Shafer Evidence Theory for Hierarchy Evaluation

In order to apply Dempster-Shafer theory for the evaluation of the classifier hierarchy it is at first necessary to derive basic probability assignments m_j from the outputs of the individual classifiers within the hierarchy. Not all neural classifiers produce output that satisfies the conditions for probability assignments (equation 1). In these cases a transformation of the outputs is necessary. The output of fuzzy k-nearest neighbour classifiers $\Xi_i(x)$ fulfils the conditions for basic probability assignments as the class memberships satisfy the conditions $\Xi_i(x) \in [0,1]$ and $\sum_{i=1}^{l} \Xi_i(x) = 1$ whereas the output of radial basis function networks $z_i(x)$ does not necessarily do so. To enforce the fulfillment of the condition $z_i(x) \in [0,1]$ a ramp function $\Theta(z_i(x)) = \begin{cases} 0, x < 0 \\ x, 0 \leq x \geq 1 \\ 1, x > 1 \end{cases}$ is applied to the classifier output setting all negative values to zero and all values greater than one to one. This is justified insofar as only a negligible number of output values violate this condition. In order to account for ignorance which is represents by

low classifier outputs the difference of the sum of the output values to one is assigned to Ω. If the sum of the classifier outputs is equal to or greater than one nothing is assigned to Ω. In this case the output is then normalised to sum up to one. Hence in either case the condition $\sum_{i=1}^{l} m_j(i) = 1$ is satisfied. These transformations are applied if necessary to the outputs of all classifiers and then the resulting basic probability assignments m_j of all classifiers are combined using the orthogonal sum without normalisation (equation 6).

According to the hierarchy structure each classifier provides evidence for the specific subsets of Ω between which the classifier discriminates and for Ω. In case of ignorance strong evidence is assigned to Ω.

Furthermore, a discounting technique is used propagating the classifier responses at higher levels of the hierarchy down. Thus classifier responses along pathes that at a higher level contain a classifier that assigned low responses are weakened strongly whereas pathes below classifiers with strong output are hardly weakened. The discounting is realised by successively multiplying the classifier responses with the classifier output of the respective predecessor node. Hence the root node is not discounted. The discounting accounts for the fact that within the hierarchy there are a not negligible number of classifier that have to provide results for samples belonging to classes they have not been trained with. Hence low classifier responses, as would be desired, cannot be guaranteed in that cases. The discounting thus weakens insular strong responses, which are likely to be caused by a classifier that has been presented a sample of an unknown class. Whereas if only one classifier within a specific path shows a low response but all other classifiers responses are high this leads only to a moderate attenuation. The discounting is applied directly after the transformation of the classifier outputs to basic probability assignments. As a multiplication with the discounting factors $d_i \in [0, 1]$ decreases the basic probability assignments if $d_i < 1$, their sum is then smaller than one $\sum_{j=0}^{s_i-1} d_i m_i(C_{i,j}) < 1$. The difference to one originating from this is then assigned to Ω: $m_i(\Omega) = 1 - \sum_{j=0}^{s_i-1} d_i m_i(C_{i,j})$.

3 Results

The proposed approach was evaluated by means of 10 runs of 10-fold cross-validation experiments on three different data sets. The data sets used were the Columbia Object Image Library (COIL20) [11] consisting of 20 objects and 72 images per object, the Letter Recognition Image Data [12] comprising 26 letters and the handwritten STATLOG digits data set [13] containing 10 digits. From the images of the COIL20 data set three different features were extracted: orientation histograms utilising sobel edge detection, orientation histograms utilising canny edge detection and wavelet coefficients.

On all three data sets the Dempster-Shafer evaluation method performed better than the simple decision-tree-like evaluation method. The average classification rates of the evidence theoretic approach are always higher than the average classification rates of the decision tree approach. The precise classification

Table 1. Classification rates for the different data sets on the test and training data for the Dempster-Shafer method (DS) and the decision tree method (DT). The evidence theoretic approach outperforms the decision tree approach in all experiments.

Data Set	Test Data		Training Data	
	DS	DT	DS	DT
Letters	$86.74 \pm 0.79\%$	$85.45 \pm 0.78\%$	$88.06 \pm 0.36\%$	$86.88 \pm 0.40\%$
Digits	$94.21 \pm 0.74\%$	$93.18 \pm 0.79\%$	$94.50 \pm 0.14\%$	$93.77 \pm 0.19\%$
COIL20	$96.88 \pm 1.48\%$	$94.56 \pm 2.04\%$	$98.88 \pm 0.23\%$	$97.74 \pm 0.30\%$

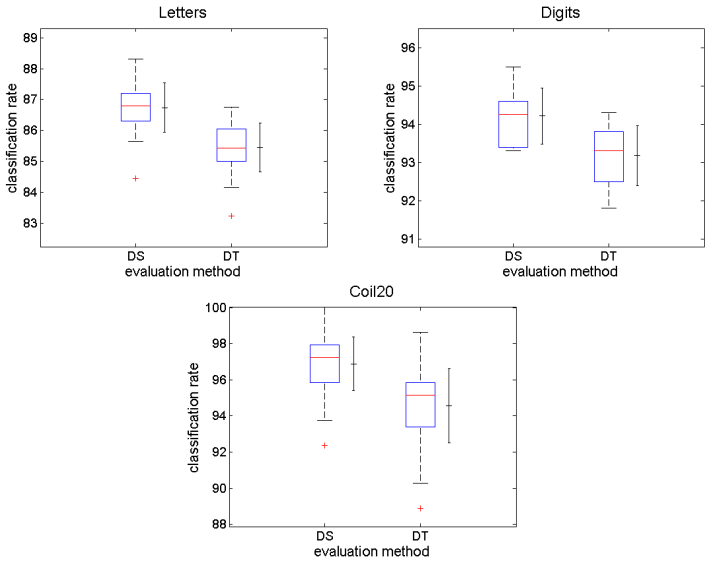

Fig. 2. Classification rates for the three data sets (letters, digits, Coil20) on the test data for the evidence based (DS) and the decision-tree-like (DT) approach. The box plots as well as the error bars indicate that Dempster-Shafer methods performs better than the decision tree method on all three data sets.

Table 2. Results of the corrected t-test for the different data sets on the test and training data comparing the Dempster-Shafer method and the decision tree method. The table gives the p-values as well as the t-value. The t-test indicates that the evidence theoretic approach outperforms the decision tree approach significantly.

Data Set	Test Data		Training Data	
	t	p	t	p
Letters	9.2753	$3.5349e-10$	20.3088	$1.0835e-18$
Digits	4.8025	$9.7038e-4$	14.5896	$1.4351e-7$
COIL20	4.7021	$8.3433e-6$	15.2295	$1.0837e-27$

rates for the different data sets can be found in table 1. Figure 2 visualises these results by means of box plots and error bars.

A pairwise t-test based on repeated k-fold cross validation with a variance correction [14] to compensate the highly violated independence assumption, called corrected repeated k-fold cross validation test, implies that the classification results for the evidence theoretic approach are significantly better than the results for the decision-tree-like approach. Table 2 contains the results of the t-test for the different data sets.

4 Discussion

The evaluation of the classifier hierarchy by means of Dempster-Shafer evidence theory yields improved classification results compared to the simple decision-tree-like evaluation method. With respect to computation time the decision tree approach outperforms the Dempster-Shafer alternative as for the former only part of the classifiers are evaluated and for the latter all classifiers within the hierarchy are used and additional calculations for transforming the classifier outputs and combining the individual classification results are needed.

Thus in time critical realtime applications an efficient approach would be to first use the simple and faster decision-tree-like method to classify the objects in question. If this method does not yield unambiguous results, the more time consuming Dempster-Shafer method should be used. If time is no critical factor, the usage of the evidence based approach is justified and recommended.

When using the evidence theoretic approach instead of the decision tree approach the advantage of the availability of intermediate classification outputs and the resulting savings of computation time do no longer apply as all classifiers within the hierarchy need to be evaluated. However, the Dempster-Shafer approach provides not only the resulting class but also a measure how likely the presented samples belongs to the specific classes.

A major drawback of the decision-tree-like evaluation method is the fact that there is no possibility to later on correct misclassifications that occur at higher levels of the hierarchy. As the evidence based approach considers all classifiers within the hierarchy a misclassification at higher levels of the hierarchy can be compensated for if the decisions made by the classifiers at the lower levels are correct. The evidence theoretic approach can only compensate misclassifications at higher levels of the hierarchy. If the misclassification takes place at a leaf node, this wrong decision cannot be corrected any more. The evidence theoretic approach can also not compensate for misclassifications where the majority of the classifiers supports the wrong decision.

5 Related Work

As the decomposition of problems into simpler sub-problems features advantages such as effectiveness and efficiency in learning and interpretability modular learning has attracted much interest recently. There are various ways of dividing a

problem into less complex sub-problems. One possible way is a partitioning of the output space. In [2] a hierarchical decomposition of a multi-class problem into several two-class problems is performed utilising Fisher discriminant analysis in combination with a deterministic annealing process. The grouping of the classes is based on the class distributions resulting in a binary tree architecture. Simple Bayesian classifiers are used to solve the sub-problems. The approach is applied to the problem of categorising landcover using hyperspectral data. Instead of Bayesian classifiers support vector machines are used in [15]. The approach has been evaluated on several pattern recognition problems. An alternative method for the decomposition of the output space is applied in [1]. A max-cut algorithm is successively applied in order to find those class partitions that have a maximal distance. As classifiers support vector machines are used. Another approach for building a hierarchical binary tree classifier architecture is proposed in [3] where a self-organising map is trained in the kernel space where classification by the deployed support vector machines takes place. On the basis of the trained self-organising map the class grouping is determined by identifying the grouping that maximises the inter-group distance while minimising the intra-group variance. In this architecture no disjoint partitioning of the classes is forced, but overlaps are allowed and are shown to improve the performance.

Dempster-Shafer evidence theory has been applied to classifier fusion in numerous applications. In [16] Dempster-Shafer theory was used for multiple classifier fusion. This approach uses prototype-based classifiers and calculates belief functions from distance measures of different classifiers which are then combined utilising Dempster-Shafer evidence theory. As distance measures the inter-class-distances and intra-class-distances were used. Classification rates, misclassification rates and rejection rates were used to derive basic probability assignments in [17]. Dempster's combination rule is applied to combine the evidences. This approach considers an extra class representing unknown classes or ignorance. In [18] a technique closely related to decision templates [19] is used to calculate degrees of belief. The distances between the classifier outputs for the sample to be classified and the mean classifier outputs calculated on the training samples are transformed into basic probability assignments. The so calculated evidences are then combined using the orthogonal sum. This approach has been varied in [20] by using reference outputs adapted to the training data so that the overall mean square error is minimised instead of simply using the mean classifier outputs. In [21] Demspter-Shafer evidence theory is used to combine the normalised outputs of multiple classifiers and to reject samples in case of highly conflicting information. If at all these approaches only exploit the possibility to allocate evidence to non-atomic hypotheses by assigning masses to atomic hypotheses θ_i and to their not necessarily atomic complement $\overline{\theta_i}$ or to the frame of discernment Ω. The proposed approach utilises this possibility as the classifier hierarchy naturally provides classification results for sets of hypotheses. Expert knowledge about the domain of application is used in [22] to calculate basic probability assignments not only for atomic hypotheses but also for composite hypotheses. Hence this approach is rather specific and less general than the proposed approach.

6 Conclusions

The proposed approach of utilising Dempster-Shafer evidence theory for the evaluation of classifier hierarchies has proven functional and shows encouraging results. It yields better classification results than the simple decision-tree-like evaluation strategy, but is more time-consuming. The already good classifications results that are achieved with a simple decision-tree-like evaluation method can be further improved using a more complex evidence based evaluation strategy. The hierarchical class grouping inherent to the classifier hierarchy seems suitable for being utilised within the framework of the Dempster-Shafer evidence theory.

Acknowledgement

This research has been partially supported by the European Union grant #IST-2001-35282 of the MirrorBot project and by the DFG (German Research Society) grant SCHW 623/3-2.

References

1. Chen, Y., Crawford, M., Ghosh, J.: Integrating support vector machines in a hierarchical output space decomposition framework. In: IEEE International Geoscience and Remote Sensing Symposium. Volume II. (2004) 949 – 952
2. Kumar, S., Ghosh, J., Crawford, M.: Hierarchical fusion of multiple classifiers for hyperspectral data analysis. International Journal on Pattern Analysis and Applications **5**(2) (2002) 210–220
3. Cheong, S., Oh, S., Lee, S.Y.: Support vector machines with binary tree architecture for multi-class classification. Neural Information Processing - Letters and Reviews **2**(3) (2004) 47–51
4. Schwenker, F.: Solving multi-class pattern recognition problems with tree structured support vector machines. In Radig, B., Florczyk, S., eds.: Mustererkennung 2001, Springer (2001) 283–290
5. Simon, S., Schwenker, F., Kestler, H.A., Kraetzschmar, G.K., Palm, G.: Hierarchical object classification for autonomous mobile robots. In: International Conference on Artificial Neural Networks (ICANN). (2002) 831–836
6. Shafer, G.: A Mathematical Theory of Evidence. University Press, Princeton (1976)
7. Dempster, A.P.: A generalization of bayesian inference. Journal of the Royal Statistical Society **B**(30) (1968) 205–247
8. Smets, P., Kennes, R.: The transferable belief model. Artificial Intelligence **66**(2) (1994) 191–234
9. Smets, P.: The combination of evidence in the transferable belief model. IEEE Transactions on Pattern Analysis and Machine Learning **12**(5) (1990) 447–458
10. Schwenker, F., Kestler, H.A., Palm, G.: Three learning phases for radial-basis-function networks. Neural Networks **14** (2001) 439–458
11. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (coil-20). Technical Report Technical Report CUCS-005-96, Columbia University (1996)

12. Frey, P.W., Slate, D.J.: Letter recognition using holland-style adaptive classifiers. Machine Learning **6**(2) (1991) 161–182
13. Kressel, U.H.G.: The impact of the learning-set size in handwritten-digit recognition. In: Proceedings of the International Confernece on Artificial Neural Networks, ICANN 1991, Elsevier Science Publishers B.V. (1991) 1685–1689
14. Bouckaert, R.R., Eibe, F.: Evaluating the replicability of significance tests for comparing learning algorithms. In: Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD 2004. Volume 3056 of LNAI., Springer (2004) 3–12
15. Rajan, S., Ghosh, J.: An empirical comparison of hierarchical vs. two-level approaches to multiclass problems. In: Multiple Classifier Systems, Proceedings of the 5th International Workshop. Volume 3077 of LNCS., Springer (2004) 283–292
16. Mandler, E., Schürmann, J.: Combining the classification results of independent classifiers based on the dempaster/shafer theory of evidence. In: Pattern Recognition and Artificial Intelligence PRAI. (1988) 381–393
17. Xu, L., Krzyzak, A., Suen, C.Y.: Methods of combining multiple classifiers and their application to handwriting recognition. IEEE Transaction on Systems, Man and Cybernetics **22**(3) (1992) 418–435
18. Rogova, G.: Combining the results of several neural network classifiers. Neural Networks **7**(5) (1994) 777–781
19. Kuncheva, L.I., Bezdek, J.C., Duin, R.P.W.: Decision templates for multiple classifier fusion: An experimental comparison. Pattern Recognition **34**(2) (2001) 299–314
20. Al-Ani, A.: A new technique for combining multiple classifiers using the dempster-shafer theory of evidence. Journal of Artificial Intelligence Research **17** (2002) 333–361
21. Thiel, C., Schwenker, F., Palm, G.: Using dempster-shafer theory in mcf systems to reject samples. In: Proceedings of the 6th International Workshop on Multiple Classifier Systems, MCS 2005. Volume 3541 of LNCS., Springer (2005) 118–127
22. Milisavljevic, N., Bloch, I.: Sensor fusion in anti-personnel mine detection using a two-level belief function model. IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews **33**(2) (2003) 269–283