A Generic Hybrid Classifier Based on Hierarchical Fuzzy Modeling: Experiments on On-Line Handwritten Character Recognition

Nicolas Ragot, Eric Anquetil
IRISA
Université de Rennes 1 / INSA de Rennes
Av. du Général Leclerc, 35042 Rennes cedex, France
Nicolas.Ragot@irisa.fr, Eric.Anquetil@irisa.fr

Abstract

In our previous works, a recognition system named ResifCar was designed specifically for on-line handwritten character recognition. This system is based on an explicit modeling by hierarchical fuzzy rules. Thus, it is understandable an optimizable after the learning stage. We present in this article a new classifier that is an extension of ResifCar. Indeed it tries to combine ResifCar's advantages with a generic aspect to handle different recognition problems. This new hybrid system combines two complementary levels. The first one uses a robust modeling by an intrinsic fuzzy clustering of each class and determines their confusing areas. The second level, based on fuzzy decision trees, operates a progressive discrimination inside these areas. Both levels are formalized by fuzzy inference systems organized hierarchically and fused for final decision. Experiments were conducted on the one hand on classical benchmarks and on the other hand on on-line handwritten digits and lower-case letters. For all of these cases, the classifier achieves good recognition rates without final optimization.

1. Introduction

Handwriting recognition has always been a complex problem for classification systems. Indeed, there are lots of different kind of symbols to recognize (letters, digits, mathematic symbols, ...) and for each of them the number of classes to identify can also be very important. Moreover, there are different allographic shapes possible for the same class: different writers will not write the same symbol in the same way and even for the same writer, differences can appear between to identical symbols. To deal with such problems and to obtain good recognition rates, an approach consists in elaborating systems specifically designed (using *a priori* knowledge) for the recognition of one category of symbols (letters, digits, ...). In previous works, we have conceived such a system. ResifCar [1] was specifically elaborated for on-line cursive character recognition using knowledge about the different possible allographic shapes and the way they are written. During the design, a great care has been taken to elaborate a "transparent" (understandable) system which could be improved after the learning phase by an expert. Thanks to these transparency properties, ResifCar can actually achieve good recognition performances for the latin letters, digits and some common symbols. Moreover, thanks to the compactness of the modeling process, it has successfully been integrated on a Smart Phone device with limited resources [1]. But the strength of the system is also its main drawback: its dedicated architecture based on *a priori* knowledge on cursive handwriting makes its adaptation difficult to other classification tasks.

The purpose of this article is to describe a new classifier which is an extension of Resifcar. The main goals are to obtain a generic system able to deal with different kinds of classification problems without needed *a priori* knowledge and without loss of transparency so that *a posteriori* optimization can be done by experts. The system must also be as compact as possible for integration on devices with limited resources such as PDA or Smart Phone.

The following section gives a quick description of ResifCar and its main properties. Then, the section 3 describes the new generic classifier based on similar theoretical concepts. Finally, the section 4 comments the different results obtained on classical benchmarks, digit recognition and lower-case letter recognition.

2. Description of ResifCar

ResifCar [1] is an on-line handwritten character recognizer which is an evolution of previous academic developments [2]. The classifier is based on an explicit and com-

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 IEEE

pact hierarchical fuzzy modeling of the knowledge guided by *a priori* knowledge on on-line cursive handwriting. These properties make the system "transparent" which allow *a posteriori* optimization.

The design of the system is done in three main steps. In a pre-processing step, the learning database is decomposed in subdatabases corresponding to the different allographic shapes of the symbols to recognize. For example, for lower-case letters and 6 special symbols, 63 subclasses where distinguished by experts. In a second step, hierarchical fuzzy models are built automatically from each subdatabases. These models are based on a three layers tree structure elaborated *a priori*. The first layer models the downstrokes which are the most robust and pertinent knowledge in cursive handwriting. The second layer operates a modeling of the morphological context of these strokes. Finally, the third layer characterizes the ligatures of the characters. Even if the structure is pre-determined, the knowledge (fuzzy sets here) for each allographic shape is automatically extracted by fuzzy clustering algorithms. In a third step, an expert can use the transparency of the system to spot problems and modify the classifier to optimize it. For example, fuzzy sets can be modified (position, size, ...), added or removed to absorb more symbols or to limit overlapping.

Thanks to the fuzzy modeling, ResifCar is able to limit the impact of noise in data and can absorb the variability of the shapes. Moreover, the "transparent" and explicit modeling had offered the possibility of decreasing the size of the system and speeding up the recognition without loss of performances. Thus ResifCar has been successfully integrated on a Smart Phone device with limited resources (memory and processor) [1].

3. A generic hybrid system

The new classifier described in this paper can be viewed has an extension of ResifCar considering that:

- it preserves a fuzzy modeling process to organize and model the knowledge in a robust, compact and comprehensive way to allow *a posteriori* optimization;
- but contrary to ResifCar, the architecture is now automatically determined so that it can be adapted to different kinds of recognition problems without needed *a priori* knowledge (generic aspect).

The system is based on an hybrid architecture composed of two complementary modeling levels. The first one determines a pre-classification according to a stable representation deduced from the intrinsic characteristics of each class to model; it is called **intrinsic level**. The second one improves the previous results by modeling relative decision boundaries between samples of a given class and samples of the others identified, according to the intrinsic modeling, as a source of confusion. This is the **discriminant level**. The two levels are built automatically by specific algorithms: a Possibilistic C-Mean algorithm [9] for the intrinsic level and fuzzy decision trees using the Fuzzy C-Mean algorithm [5] for the discriminant level. Both allow an homogeneous and understandable formalization by fuzzy inference systems.

The next subsections describe the learning of the two levels and their combination for the recognition of unknown symbols.

3.1. Intrinsic level

The purpose of this first level is to model each class to be recognized according to its intrinsic characteristics. Using this kind of modeling, the system is able to find automatically stable subclasses that compose the initial classes. This extraction of subclasses can be compare to the preprocessing step that is done for ResifCar where the different possible allographs of a same symbol are selected *a priori* for the learning of the classifier (cf. section 2). The intrinsic models (description of the subclasses) are generated by an adapted fuzzy clustering algorithm like the Possibilistic C-Mean (PCM) [9] on each class separately. In this way, for a given class Ci, the corresponding model MCi is composed of a set of L fuzzy sets FS^l_{Ci} $(l = 1, \ldots, L)$ whose membership functions are $\mu_{FS^l_{Ci}}$ (cf. fig. 1).

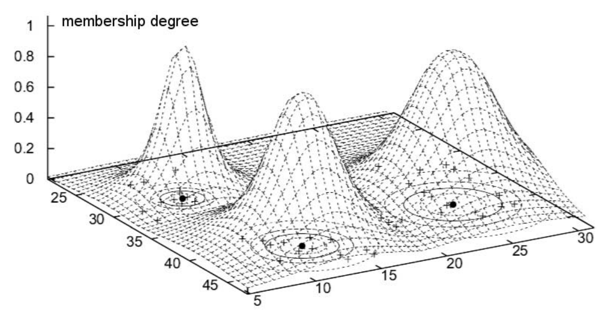

Figure 1. Example of the intrinsic modeling of a class determined by a PCM algorithm.

The PCM algorithm has been chosen for its interesting property: clusters are based on the notion of "typicality" that corresponds to the intrinsic characteristics searched (unlike fuzzy c-mean for example where the clusters are much more based on the notion of "sharing" (cf. section 3.2)). The resulting fuzzy modeling is understandable and can absorb variability of the symbols.

Once the models are extracted for each class, they are used to design a Fuzzy Inference System (FIS) that deter-

0-7695-1960-1/03 $17.00 © 2003 IEEE

mines the pre-classification. Each rule RCi of the FIS describes a whole class Ci:

$\boldsymbol{R_{Ci}}$: **If** symbol match MCi **THEN** its class is Ci.

The premise corresponds to the membership degree $\mu_{MCi}(e)$ of a symbol e to the intrinsic model of the class Ci. It is calculated using the T-conorm *max* by :

$$\mu_{MCi}(e) = \max_{l=1,\dots,L} (\mu_{FS^l_{Ci}}(e)). \tag{1}$$

The conclusion are formulated classically in a numerical manner and are improved by a pseudo-inverse algorithm.

During inference, the rules of this FIS are aggregated by a *sum-product* inference that assigns a score $P1_{Ci}(e)$ to each class, denoting the similarity between the input symbol e and the intrinsic model of the class. Ranking these scores determines the pre-classification.

Another interesting property of this intrinsic modeling is that "typicality" provides noise immunity and allow to reject symbols that don't match none of the classes. For example, it is possible to determine thresholds Rej_{Ci} for each class Ci. If the membership degree $\mu_{MCi}(e)$ of a symbol e is less than Rej_{Ci}, e cannot belong to Ci and if e cannot belong to any classes then e is rejected. This kind of reject improves the robustness of the classification process and is very useful when severals classifiers are used at the same time (by cascading or combining them) for specific tasks (for example, one classifier can be used for the recognition of digits, another for lower-case letters and another for upper-case letters).

3.2. Discriminant level

We have shown that the first level of the hybrid system operates in the same way than ResifCar do: both model subclasses of the initial classes. ResifCar's modeling is based on an *a priori* decomposition in allographic shapes, whereas the generic hybrid system determines automatically the subclasses according to intrinsic characteristics of the classes.

In a second step, ResifCar operates a fuzzy modeling for each allograph to discriminate them from each other. This modeling is based on a tree structure determined *a priori*. It is composed of three layers using knowledge from the more pertinent and stable one to the more sensible one. Similarly, the aim of the second level of the hybrid system is to operate a progressive discrimination of the classes using the most pertinent knowledge first. To do this, Fuzzy Decision Trees (FDT) are built automatically to find decision boundaries between symbols of a given class and others that are confusing according to the intrinsic level.

Firstly, the intrinsic level is used to determine for each class which kind of symbols (this not necessarily whole classes) are confusing. Thus, for each class Ci, a learning database BCi is composed with the symbols that potentially belong to Ci. These symbols e satisfy the following condition:

$$\mu_{MCi}(e) \geq \frac{\max_k(\mu_{MCk}(e))}{\alpha}. \tag{2}$$

Then, for the discrimination, symbols that really belong to Ci are considered as positive examples whereas the others are considered as counter-examples.

To find the decision boundaries, FDT are built from each database BCi. This modeling process has some interesting properties [8, 11, 12]. It is able to select automatically (using a criterion like the star entropy [13]) the best feature subspace for local discrimination; it can absorb the variability of the symbols thanks to the fuzzy representation; and more than all, it can be easily formulated as FIS, which produces an homogeneous representation with the intrinsic level. The novelty of the FDT used [12] lies in the fact that the partitionings are done by the Fuzzy C-Mean algorithm (FCM) [5]. Indeed, thanks to the properties of this algorithm, each symbol has a degree of "sharing" between the different clusters found. Thus, the corresponding fuzzy sets (that are automatically extracted) determine implicit fuzzy boundaries useful for the discrimination (cf. fig. 2).

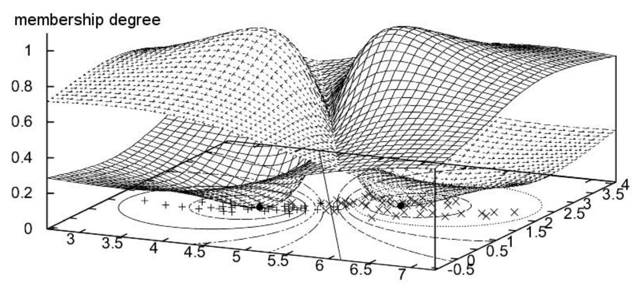

Figure 2. Example of discriminant modeling determined by the FCM.

Each FDT is used to deliver a score $P2_{Ci}$ indicating how an input symbol belong to the class the FDT was designed for (remind that each tree is built from a database BCi corresponding to a class Ci). These scores are obtained by infering the FIS deduced from the trees [12].

3.3. Collaboration of the intrinsic level and discriminant level for the decision

The two levels of the hybrid system are based on complementary modeling. The first one model stable knowledge based on intrinsic characteristics of each of the classes and establish a pre-classification. The second one uses this intrinsic modeling to decompose the initial problem in subproblems (confusing areas of each class) and operates a local discrimination on each one to reduce pre-classification errors.

During the generalization phase (classification of unknown symbols), this complementarity is used to take the decisions. Firstly, the FIS of the intrinsic level is infered to determine the scores $P1_{Ci}(e)$ of the input symbol e. Then, the symbol activates the FDT of the discriminant level corresponding to the classes that satisfy the condition (2). The associated FIS are then used to obtained the scores $P2_{Ci}(e)$. For the FIS that are not activated (because the corresponding class do not satisfy the condition (2)), $P2_{Ci}(e)$ is fixed to 0. The final decision $G_{Ci}(e)$ can then be taken by an appropriate combination of the scores of the two levels: $G_{Ci}(e) = P1_{Ci}(e) \times P2_{Ci}(e)$ and selecting the class Ci for which G_{Ci} is maximum. In this combination, the product is used for its conjunction property: a sample must belong to the class Ci at the intrinsic level AND at the discriminant level at the same time to validate the class Ci.

4. Experimental results

To evaluate this generic hybrid system and its aptitude to adapt itself to several classification problems, different experiments were done on classical benchmarks and on on-line character recognition (digits and lower-case letters).

For classical benchmarks, the classifier has been evaluated on two databases of the UCI Repository[1]: the Breiman's waveform and the StatLog satellite images. For the waveform, 600 samples are used for learning and 3000 others for the tests. For the satellite images, the given datasets are used (4435 samples for learning and 2000 others for tests). The table 1 summarizes the results on test sets for the generic hybrid system and three of the best classification algorithms on these databases among the 33 evaluated in [10]. In comparison with these 33 algorithms, the presented system obtain good performances in the same test conditions.

Table 1. Recognition rates of several classifiers on two classical benchmarks.

	Waveforms	Satellite images
Generic hybrid system	85.2 %	89.6 %
LVQ	83.0 %	90.2 %
RBFN	84.9 %	87.9 %
Quest	82.3 %	84.6 %

Actually, preliminary experiments are carried out on on-line character recognition. The main goal is to evaluate the initial performances (without *a posteriori* optimization and without use of reject) of the generic hybrid classifier on digits and lower-case letters recognition. To have a good comparison, tests were done in the same conditions with a Support Vector Machine classifier (SVM) [6] which is known as being one of the most powerful classifier of the moment. The SVM was trained in the multiclass mode (one class against the others) using gaussian kernels and with different sets of parameters. No comparison is reported with ResifCar (results can be found in [1]) because it was specifically optimized *a posteriori* by an expert to satisfy particular conditions for its integration on a Smart Phone device.

For on-line digit recognition, experiments where done on the IRONOFF [14] database using 50% for learning and the other 50% for test. There are around 400 writers in the database and they are different in the learning set and in the test set. There is no pre-processing of the digits which are directly described by a set of only 43 high level features similar to those used for ResifCar [1, 2].[2]

Table 2. Comparison of the performances of the presented generic hybrid system and a SVM classifier on on-line digit recognition.

	Generic hybrid system	SVM
Recognition rates	95.6 %	95.5 %
Nb of parameters	12 150	114 075

As we can see in table 2, these preliminary results on the test set are good but not as good as those expected. This can be explain essentially by the set of features used which is not refined enough. Indeed, the SVM classifier gives similar results. However, these experiments highlights an interesting property of the generic hybrid system: thanks to the fuzzy modeling, it can achieve similar recognition rates than the SVM with about 10 times less parameters.

Last experiments were carried out on on-line lower-case letter recognition using the section $1c$ of the UNIPEN (train r01-v07) database [7]. The characters are described by an extended set of 73 high level features similar to those previously used for digit recognition.[2] No pre-processing and no cleaning of the database is done except for characters from which features can't be extracted (241 of the 61351). The data are divided randomly into two disjoint sets: $2/3$ for learning and $1/3$ for test. Preliminary results on test set are reported in table 3 where the system is compared on the same databases with the SVM.

These results are encouraging because the UNIPEN database is a difficult benchmark with bad characters (bad segmented, bad labeled, poor resolution, ...). Since the experiments were done without pre-processing, these problems can disturb the learning (notably fuzzy clustering and

[1] http://www.ics.uci.edu/∼mlearn/MLRepository.html

[2] We cannot provide more informations on these features here for contractual reasons.

Table 3. Performances of a SVM and the generic hybrid classifier on on-line lower-case letter recognition.

	Generic hybrid system	SVM
Recognition rates	89.7 %	92.2 %
Nb of parameters	82 506	2 438 784

optimization of the conclusion of the FIS which is done by the pseudo-inverse algorithm). It is difficult to find in the literature, experiments that were done in similar conditions but they seem to be close to those obtained. For example, a Hidden Markov Model classifier based on Statistical Dynamic Time Warping (SDTW) [3] achieves 90.3 % of recognition rate (results reported from [4]). However, results of the SVM seem to show that it must be possible to improve the generic hybrid system either by refining the initial parameters (α in condition (2), clustering parameters, ...) and/or by *a posteriori* optimization. These optimizations must also allow to decrease significantly the number of parameters (as it was for ResifCar) which is yet about 30 times less than those needed for the SVM. Thus, the compactness of the system makes its integration on "small" devices such as PDA or smart phone easier.

5. Conclusion

Our previous works on handwriting recognition and notably experiences based on ResifCar's integration on a Smart Phone device convinced ourself that transparency in classification system is a powerful strength. Indeed, it provides modularity to the system and gives the opportunity to an expert to modify parts of the classifier according to spotted problems after learning stage.

In this article we have presented a new generic hybrid classifier which purpose is to provide not only transparency but also a generic aspect for automatic adaptation to several classification tasks. The hybrid system uses a double modeling based on an intrinsic representation of the classes and a discriminant part to refine the modeling near decision boundaries. The whole system is formalized by fuzzy inference systems for their compactness, comprehensibility and their ability to absorb variability of inputs.

Tests were done on classical benchmarks and on on-line character recognition. In all cases, results are close to other approaches among the best which is promising since all the capacities of the system were not exploited. Moreover, the compactness of the generic hybrid system is far more important than the one of a SVM which can allow integration on low resources devices. In future works, some improvements are envisaged like introducing reject management, working with a validation dataset and using transparency for *a posteriori* optimization. Possibilities of integration on PDA will also be studied.

References

[1] E. Anquetil and H. Bouchereau. Integration of an on-line handwriting recognition system in a smart phone device. In *sixteenth IAPR International Conference on Pattern Recognition (ICPR 2002)*, volume III, pages 192–195, 2002.

[2] E. Anquetil and G. Lorette. Automatic generation of hierarchical fuzzy classification systems based on explicit fuzzy rules deduced from possibilistic clustering: Application to on-line handwritten character recognition. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'96)*, pages 259–264, 1996.

[3] C. Bahlmann, B. Haasdonk, and H. Burkhardt. Measuring HMM similarity with the bayes probability of error and its application to online handwriting recognition. In *Proceedings of the 6th ICDAR*, pages 406–411, 2001.

[4] C. Bahlmann, B. Haasdonk, and H. Burkhardt. On-line handwriting recognition with support vector machines - a kernel approach. In *Proceedings of the 8th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pages 49–54, 2002.

[5] J. C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Plenum Press, 1981.

[6] R. Collobert and S. Bengio. SVMTorch: Support vector machines for large-scale regression problems. *Journal of Machine Learning Research*, 1:143–160, 2001.

[7] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet. UNIPEN project of on-line data exchange and recognizer benchmarks. In *Proceedings of the 12th ICPR*, pages 29–33, 1994.

[8] C. Z. Janikow. Fuzzy decision trees: Issues and methods. *IEEE Transactions on Systems, Man and Cybernetics*, 28:1–14, 1998.

[9] R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, 1993.

[10] T.-S. Lim, W.-Y. Loh, and Y.-S. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40(3):203–228, 2000.

[11] C. Marsala and B. Bouchon-Meunier. An adaptable system to construct fuzzy decision trees. In *18th International Conference of the North American Fuzzy Information Processing Society (NAFIPS)*, pages 223–227, 1999.

[12] N. Ragot and E. Anquetil. A new hybrid learning method for fuzzy decision trees. In *Proceedings of the 10th IEEE International Conference on Fuzzy Systems*, volume 3, 2001.

[13] H. Tanaka, T. Okuda, and K. Asai. Fuzzy information and decision in statistical model. *Advances in Fuzzy Set Theory and Applications*, pages 303–320, 1979.

[14] C. Viard-Gaudin, P. M. Lallican, S. Knerr, and P. Binter. The IRESTE on/off (IRONOFF) dual handwriting database. In *Proceedings of the Fifth International Conference on Document Analysis and Recognition (ICDAR'99)*, pages 455–458, 1999.

0-7695-1960-1/03 $17.00 © 2003 **IEEE**