Fuzzy-Pattern-Classifier Training with Small Data Sets

Uwe Mönks[1], Denis Petker[2], and Volker Lohweg[1]

[1] inIT – Institute Industrial IT, Ostwestfalen-Lippe University of Applied Sciences
Liebigstr. 87, D-32657 Lemgo, Germany
{uwe.moenks,volker.lohweg}@hs-owl.de
[2] OWITA GmbH, Langenbruch 23, D-32657 Lemgo, Germany
denis.petker@owita.de

Abstract. It is likely in real-world applications that only little data is available for training a knowledge-based system. We present a method for automatically training the knowledge-representing membership functions of a Fuzzy-Pattern-Classification system that works also when only little data is available and the universal set is described insufficiently. Actually, this paper presents how the Modified-Fuzzy-Pattern-Classifier's membership functions are trained using probability distribution functions.

Keywords: Fuzzy Logic, Probability Theory, Fuzzy-Pattern-Classification, Machine Learning, Artificial Intelligence, Pattern Recognition.

1 Introduction

In many knowledge-based industrial application there is a necessity to train using a small data set. It is typical that there are less than ten up to some tens of training examples. Having only such a small data set, the description of the underlying universal set, from which these examples are taken, is very vague and connected to a high degree of uncertainty. It was Zadeh [1] who created the basic theory for the nowadays established fuzzy systems, which are suitable for modelling uncertain knowledge using possibility measures. One class of such systems are the *Fuzzy-Pattern-Classifiers (FPC)* introduced by Bocklisch [2] which are widely used in pattern recognition applications for object classification. The basic concept is having a set of *fuzzy membership functions* $\mu : x \rightarrow [0, 1]$ per class which model characteristic features of those classes. These membership functions map an object's feature value $x \in \mathbb{R}$ to the unit interval representing the membership or degree of similarity of x to an ideal class member's feature. All memberships are aggregated subsequently by some *fuzzy aggregation operator*. The object is then assigned to the class having the highest aggregated value.

One established member of the class of Fuzzy-Pattern-Classifiers is Lohweg's *Modified-Fuzzy-Pattern-Classifier (MFPC)* [3,4]. It is widely applied and established in the industry, for instance in printing facilities for checking the print results to give only one example [4,5,6,7]. In these applications, it proved its

E. Hüllermeier, R. Kruse, and F. Hoffmann (Eds.): IPMU 2010, Part I, CCIS 80, pp. 426–435, 2010.

robustness, performance, and efficiency when implemented in hardware-based solutions.

The MFPC's membership functions are parameterisable unimodal potential functions having at least two degrees of freedom left to the user which demand the application of costly heuristics for finding their values. Mostly, the optimal parameters are not found, resulting in a loss of robustness and therefore deteriorated classification rates.

In this paper we suggest an automatic method of learning the fuzzy membership functions by estimating the data set's probability distribution and deriving the function's parameters automatically from it. The resulting *Probabilistic MFPC (PMFPC)* membership function, extends the MFPC approach to asymmetric membership functions and leaves only one degree of freedom leading to a shorter learning time for obtaining stable and robust classification results.

There exist other approaches in the literature, which go in our direction, but are not applicable here. Rodner and Denzler's approach [8] transfers feature relevance from previous, similar applications to choose the respective features for a new classification task. Our approach is directed to applications where no previous knowledge is available and the features are chosen heuristically. Drobics et al.'s FS-FOIL method is also very promising, but the classification results presented in [9] make use of a bigger training set. Also, the learning approach of finding fuzzy decision rules is different from ours where fuzzy membership functions' shapes are determined.

After having introduced the topic of this paper in this Section, we proceed in Sect. 2 by briefly introducing the Modified-Fuzzy-Pattern-Classifier. In Sect. 3 the new probabilistic parameterisation approach is described. The experiments presented in Sect. 4 return promising results that the incorporation of PMFPC membership functions in fuzzy classification tasks can improve classification results significantly when compared to MFPC. The paper concludes with Sect. 5 and provides an outlook on further research.

2 Modified Fuzzy Pattern Classifier

A hardware optimized derivate of Bocklisch's *Fuzzy-Pattern-Classifier (FPC)* [2] is the *Modified-Fuzzy-Pattern-Classifier (MFPC)*, which can be efficiently implemented as a pattern recognition system on a Field Programmable Gate Array (FPGA), applicable in high-speed industrial applications [4]. Here, its properties shall be briefly introduced. For details, we refer to [4] and [6].

The hardware efficient membership function used for the MFPC is Eichhorn's parameterisable unimodal potential function [10] defined as

$$\mu_{\mathrm{MFPC}}\left(m,\mathbf{p}\right) = 2^{-d(m,\mathbf{p})} \in [0,1] \text{ with } d\left(m,\mathbf{p}\right) = \left(\frac{|m-S|}{C}\right)^{D}, \qquad (1)$$

where $\mathbf{p} = (S, C, D)$ is a parameter vector defining the membership function's properties, namely mean value (S), width (C), and steepness of its edges (D). $d\left(m,\mathbf{p}\right)$ is the distance measure of the inspected feature m with regard to the

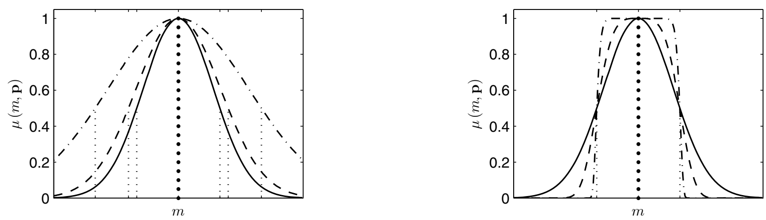

Fig. 1. Sample MFPC membership function at $D = 2$ and $p_{C_e} = 0$ (*solid*). The left and right plots show changes (*dashed* $\rightarrow$ *dash-dotted*) with increasing p_{C_e} and D, respectively. The vertical *dotted* line shows respective $S \pm C$, the *bold-dotted* line S.

properties of the membership function, i. e. how far is the measured feature m away from its mean value S. A sample MFPC membership function is depicted in Fig. 1.

The MFPC membership function's parameters S and C are obtained automatically during a learning phase after extracting all regarded features m from N typical members of a class by [3] $S = \Delta + m_{\min}, C = (1 + 2p_{c_e}) \cdot \Delta$, where $p_{c_e} \in [0, 1]$ is called percental elementary fuzziness and defines an arbitrary, user-defined width adjustment factor, and where $m_{\max} = \max_{i=1}^{N} m_i, \quad m_{\min} = \min_{i=1}^{N} m_i, \quad \Delta = \frac{m_{\max} - m_{\min}}{2}$. The integer-valued parameter D is chosen arbitrarily, typically as a power of 2 to keep calculating the distance measure $d\,(m, \mathbf{p})$ hardware-efficient [3].

The MFPC aggregation of M different features is expressed by

$$h_{\text{MFPC}}\,(\mathbf{m}, \mathbf{P}) = 2^{-\frac{1}{M}\sum_{i=1}^{M} d_i(m_i, \mathbf{p}_i)}, \text{ with } d_i\,(m_i, \mathbf{p}_i) = \left(\frac{|m_i - S_i|}{C_i}\right)^{D_i}, \quad (2)$$

where $\mathbf{m}$ is a vector of feature values m_i and $\mathbf{P}$ a matrix of parameter vectors $\mathbf{p}_i$, parameterising each membership function belonging to a feature m_i. It is proved in [6] that the membership functions are aggregated using the well-known *geometric mean* aggregation operator, which is a fuzzy *averaging operator*. Since (2) can be rewritten to

$$h_{\text{MFPC}}\,(\mathbf{m}, \mathbf{P}) = \left(\prod_{i=1}^{M} 2^{-d_i(m_i, \mathbf{p}_i)}\right)^{\frac{1}{M}} = \left(\prod_{i=1}^{M} \mu_{\text{MFPC},i}(m_i, \mathbf{p}_i)\right)^{\frac{1}{M}},$$

it is possible to use any other fuzzy membership function instead of μ_{MFPC} for existing MFPC applications [6]. μ_{MFPC}'s parameters D and p_{c_e} are not determined automatically and left to the user. An appropriate substitute of μ_{MFPC}, which is parameterised completely automatically (or at least with a smaller number of free parameters) and yields optimal performances, was therefore searched for and found in the *Probabilistic MFPC* membership function. This approach is presented in the following.

3 Probabilistic MFPC Membership Function

To learn a fuzzy membership function automatically, *Random Fuzzy Variables (RFV)* can be applied [11], but this approach has disadvantages towards high-speed real-time applications. The *Probabilistic MFPC (PMFPC)* membership function approach we present here is able to preserve real-time demands (experiments revealed that the parameterisation is executed one order of magnitude faster than the RFV approach) while producing an optimal data set representation by incorporating an estimated probability distribution of the data.

The PMFPC approach is based on a generalised MFPC membership function

$$\mu_{\mathrm{PMFPC}}\left(m,\mathbf{p}\right)=2^{-\mathrm{ld}\left(\frac{1}{\mathrm{B}}\right)d(m,\mathbf{p})}\in[0,1] \text{ with } d\left(m,\mathbf{p}\right)=\left(\frac{|m-S|}{C}\right)^{D}, \quad (3)$$

where $B\in(0,1]$ is the *class boundary membership* parameter, i. e. defining the membership function's value at $m=S\pm C$: $\mu_{\mathrm{PMFPC}}\left(S\pm C,\mathbf{p}\right)=B$. This parameter was actually already introduced by Bocklisch in his Fuzzy-Pattern-Classifier definition [2]. D and B are automatically parameterised in the PMFPC approach. p_{c_e} is yet not automated to preserve the possibility of adjusting the membership function slightly without needing to learn the membership functions from scratch. The algorithms presented in this paper for automatically parameterising parameters D and B are inspired by former approaches: Bocklisch as well as Eichhorn developed algorithms which allow obtaining a value for the (MFPC) potential function's parameter D automatically, based on the used training data set. Bocklisch also proposed an algorithm for the determination of B. For details we refer to [2] and [10]. However, these algorithms yield parameters that do not fulfil the constraints connected with them (cf. Sect. 3.1 and 3.2) in all practical cases. Hence, we propose a probability theory-based alternative described in the following.

3.1 Automatically Parameterising the Steepness of the Edges

Bocklisch formulated constraints for D so the resulting membership function appropriately describes the data set for which the membership function is created [2]. He demands (i) $2\leq D\leq 20$; (ii) if the objects in the data set are uniformly distributed, the membership function should be sharp-edged ($D=20$); (iii) in case of an accumulation of objects at the outer boundaries, this distribution is represented by a sharp membership function as well ($D=20$); and (iv) an inner accumulation of objects should generate a fuzzy membership function, thus $D\to 2$. These constraints are visualised in Fig. 2, showing ten distributed data points X_i and the resulting membership function $\mu(x)$ in accordance to the aforementioned constraints. Bocklisch's and Eichhorn's algorithms adjust D after comparing the actual distribution of objects to a perfect uniform distribution. However, the algorithms tend to change D for every (small) difference between the actual distribution and a perfect uniform distribution. This explains why both algorithms do not fulfil the constraints when applied to random uniform distributions.

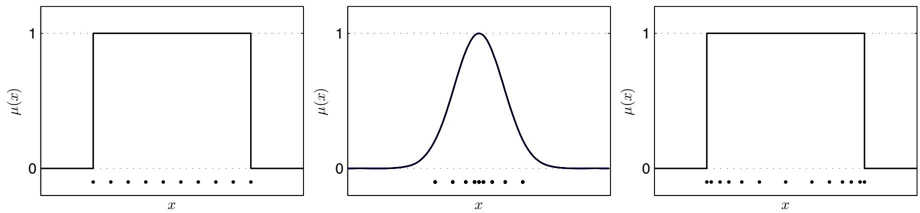

Fig. 2. Distributions of X_i (*bold dots*) and their corresponding membership functions

We actually stick to the idea of adjusting D with respect to the similarity of the actual distribution compared to an artificial, ideal uniform distribution, but we use probability theoretical concepts. Our algorithm basically works as follows: At first, the *empirical cumulative distribution function (ECDF)* of the data set under investigation is determined. Then, the ECDF of an artificial perfect uniform distribution in the range of the actual distribution is determined, too. The similarity between both ECDFs is expressed by its correlation factor which is subsequently mapped to D by a parameterisable function.

Determining the Distributions' Similarity. Consider a sorted vector of n feature values $\mathbf{m} = (m_1, m_2, \ldots, m_n)$ with $m_1 \leq m_2 \leq \ldots \leq m_n$, thus $m_{\min} = m_1$ and $m_{\max} = m_n$. The corresponding ECDF $P_m(x)$ is determined by $P_m(x) = \frac{|\tilde{\mathbf{m}}|}{n}$ with $\tilde{\mathbf{m}} = (m_i | m_i \leq x \; \forall i \in \mathbb{N}_n)$, where $|\mathbf{x}|$ denotes the number of elements in vector $\mathbf{x}$ and $\mathbb{N}_n = [1, 2, \ldots, n]$. The artificial uniform distribution is created by equidistantly distributing n values u_i, hence $\mathbf{u} = (u_1, u_2, \ldots, u_n)$ with $u_i = m_1 + (i-1) \cdot \frac{m_n - m_1}{n-1}$. Its ECDF $P_u(x)$ is determined analogously by substituting $\mathbf{m}$ with $\mathbf{u}$. In the next step, the similarity between both distribution functions is computed by calculating the *correlation factor* [12]

$$c = \frac{\sum_{i=1}^{k} \left(P_m[x_i] - \overline{P_m}\right)\left(P_u[x_i] - \overline{P_u}\right)}{\sqrt{\sum_{i=1}^{k} \left(P_m[x_i] - \overline{P_m}\right)^2 \sum_{i=1}^{k} \left(P_u[x_i] - \overline{P_u}\right)^2}},$$

where $\overline{P_a}$ is the mean value of $P_a(x)$, computed as $\overline{P_a} = \frac{1}{k} \sum_{i=1}^{k} P_a[x_i]$. c's properties can be found in [12]. It is actually the empirical correlation coefficient, demanding sampled data to be determined, necessarily sampled at the same locations x_i. Since $P_m(x)$ cannot be predicted, it seems to be appropriate to sample at k equidistantly spaced locations. k is determined by $k = 10^{\lfloor \log_{10} n \rfloor + 1}$, but at least $k = 50$. This guarantees that the functions are sampled at not less than five times as many sampling points as feature values are available. The equidistant locations are determined as $x_i = m_1 + (i-1) \cdot \frac{m_n - m_1}{k-1} \; \forall i \in \mathbb{N}_k$.

The correlation factor must now be mapped to D while fulfilling Bocklisch's constraints on D. Therefore, the average influence $\overline{\alpha}(D)$ of the parameter D on the membership function μ_{MFPC}, which is the base for μ_{PMFPC}, is investigated to derive a mapping based on it. $\alpha_D(x)$ is determined by taking $\frac{\partial}{\partial D} \mu_{\text{MFPC}}(x, D)$ with $x = \frac{m-S}{C}, x > 0$:

$$\alpha_D(x) = \frac{\partial}{\partial D}\mu_{\text{MFPC}}(x, D) = \frac{\partial}{\partial D}2^{-x^D} = \ln(2)\left(-2^{-x^D}\right)x^D\ln(x).$$

The locations x represent the distance to the membership function's mean value S, hence $x = 0$ is the mean value itself, $x = 1$ is the class boundary $S + C$, $x = 2$ twice the class boundary and so on. The average influence of D on the membership function is evaluated for $-1 \le x \le 1$: This interval bears the most valuable information since all feature values of the objects in the training data set are included in this interval, and additionally those of the class members are expected here during the classification process, except from only a typically neglectable number of outliers. Anyway, the range of x must be necessarily bounded since the average influence of D on the membership function, namely $\overline{\alpha}(D) = \frac{1}{x_r - x_l}\int_{x_l}^{x_r}\alpha_D(x)\,dx$, is computing $\alpha_D(x)$'s mean value along x. But since $\lim_{x\to\infty}\alpha_D(x) = 0\ \forall D$, integration of $\alpha_D(x)$ over $\mathbb{R}$ would yield $\overline{\alpha}(D) = 0\ \forall D$, which is not true for that range of x where the majority of objects is present. The mapping of $D : c \to [2, 20]$, which is derived in the following, must take D's average influence into consideration. A graphical representation of $\overline{\alpha}(D)$ is shown in Fig. 3 for the range $2 \le D \le 20$, which is actually the only of interest.

Mapping the Distribution's Similarity to the Edge's Steepness. In the general case, the correlation factor c can take values from the interval $[-1, 1]$, but when evaluating distribution functions, the range of values is restricted to $c \in [0, 1]$, which is because probability distribution functions are monotonically increasing. This holds for both distributions, $P_m(x)$ as well as $P_u(x)$. It follows $c \ge 0$. The interpretation of the correlation factor is straight forward. A high value of c means that the distribution $P_m(x)$ is close to a uniform distribution. If $P_m(x)$ actually was a uniform distribution, $c = 1$ since $P_m(x) = P_u(x)$. According to Bocklisch, D should take a high value here. The more $P_m(x)$ differs from a uniform distribution, the more $c \to 0$, the more $D \to 2$. Hence, the mapping function $D(c)$ must necessarily be an increasing function with taking the exponentially decreasing average influence of D on the membership function $\overline{\alpha}(D)$ into consideration (cf. Fig. 3). An appropriate mapping $D : c \to [2, 20]$ is an exponentially increasing function which compensates the changes of μ_{MFPC} with respect to changes of c. While big changes in small c values result in minor

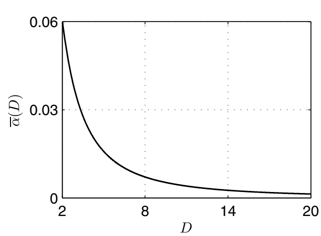

Fig. 3. Average influence of the parameter D on μ_{MFPC} with respect to D

changes of D, implying only a small change of the membership function, D increases rapidly for big correlation factors, not affecting the membership function strongly. We suggest the following heuristically determined exponential function, which achieved promising results during experiments (cf. Sect. 4):

$$D(c) = 19^{c^{2q}} + 1 \;\Rightarrow\; D(c) \in [2, 20], \tag{4}$$

where q is an adjustment parameter. This formulation guarantees that $D \in [2, 20]$ $\forall c$ since $c \in [0, 1]$. Using the adjustment parameter q, D is adjusted with respect to the aggregation operator used to fuse all n membership functions representing each of the n features. Each fuzzy aggregation operator behaves differently. For a fuzzy averaging operator $h(\mathbf{a})$, Dujmović' introduced the objective measure of *global andness* ρ_g (for details cf. [13,6]). Assuming $q = 1$ in the following cases, it can be observed that, when using aggregation operators with a global andness $\rho_g^{h(\mathbf{a})} \to 0$, the aggregated single, n-dimensional membership function is more fuzzy than that one obtained when using an aggregation operator with $\rho_g^{h(\mathbf{a})} \to 1$, where the resulting function is sharp. This behaviour should be compensated by adjusting D in such a way, that the aggregated membership functions have comparable shapes: at some given correlation factor c, D must be increased if ρ_g is high and vice versa. This is achieved by mapping the aggregation operator's global andness to q, hence $q : \rho_g \to \mathbb{R}$. Our suggested solution is a direct mapping of the global andness to the adjustment parameter q, hence $q(\rho_g) = \rho_g \;\Rightarrow\; q \in [0, 1]$. Mapping (4) is now completely defined and consistent with Bocklisch's constraints and our observations regarding the aggregation operator's andness.

3.2 Determining the Class Boundary Membership Parameter

In addition to the determination of D, we present an algorithm to automatically parameterise the class boundary membership B. This parameter is a measure for the membership $\mu_{\text{MFPC}}(m, \mathbf{p})$ at the locations $m \in \{S + C,\ S - C\}$. Typically, the class boundary membership is assigned a value of $B = 0.5$. The algorithm for determining B is based on the algorithm Bocklisch developed [2], but was not adopted as it stands since it has some disadvantages if this algorithm is applied to distributions with a high density especially on the class boundaries. Due to space limitations, this cannot be presented here.

When looking at μ_{MFPC}, the following two constraints on B can be derived: (i) The probability of occurrence is the same for every object in uniform distributions, also on the class boundary. Here, B should have a high value. (ii) For distributions where the density of objects decreases when going towards the class boundaries B should be assigned a small value, since the probability that an object occurs at the boundary is smaller than in the centre.

Hence, for sharp membership functions ($D \to 20$) a high value for B should be assigned, while for fuzzy membership functions ($D \to 2$) the value of B should be low. $B = f(D)$ must have similar properties like $\overline{\alpha}(D)$, meaning B changes quickly where $\overline{\alpha}(D)$ changes quickly and vice versa. We adopted Bocklisch's suitable equation for computing the class boundary membership [2]:

$$B = \frac{1}{1 + \left(\frac{1}{B_{\max}} - 1\right) \cdot \left(\frac{D_{\max}}{D}\right)^{1+\frac{1}{q}}},$$

where $B_{\max} \in (0, 1)$ stands for the maximum possible value of B with a proposed value of 0.9, $D_{\max} = 20$ is the maximum possible value of D and q is identical in its meaning and value to q as used in (4).

3.3 An Asymmetric PMFPC Membership Function Formulation

A data set may be represented better if the membership function was formulated asymmetrically instead of symmetrically as is the case with (3). This means

$$\mu_{\text{PMFPC}}(m, \mathbf{p}) = \begin{cases} 2^{-\text{ld}\left(\frac{1}{B_l}\right)\left(\frac{|m-S|}{C_l}\right)^{D_l}}, & m \leq S \\ 2^{-\text{ld}\left(\frac{1}{B_r}\right)\left(\frac{|m-S|}{C_r}\right)^{D_r}}, & m > S \end{cases}, \tag{5}$$

where $S = \frac{1}{M}\sum_{i=1}^{M} m_i, m_i \in \mathbf{m}$ is the *arithmetic mean* of all feature values. If S was computed as introduced in (2), the resulting membership function would not describe the underlying feature vector $\mathbf{m}$ appropriately for asymmetrical feature distributions. A new computation method must therefore also be applied to $C_l = S - m_{\min} + p_{C_e} \cdot (m_{\max} - m_{\min})$ and $C_r = m_{\max} - S + p_{C_e} \cdot (m_{\max} - m_{\min})$ due to the change to the asymmetrical formulation. To compute the remaining parameters, the feature vector must be split into the left side feature vector $\mathbf{m}_l = (m_i | m_i \leq S)$ and the one for the right side $\mathbf{m}_r = (m_i | m_i \geq S)$ for all $m_i \in \mathbf{m}$. They are determined following the algorithms presented in the preceding Sections, but using only the feature vector for one side to compute this side's respective parameter.

4 Experimental Results on PMFPC

In order to evaluate the classification performance of our probabilistic approach on parameterising the fuzzy membership functions, the same data set is used to learn both the original MFPC membership function μ_{MFPC} and also μ_{PMFPC}. This data set "OCR" (the same as is used in [6]) was compiled in an industrial optical character recognition application and consists of both a training and a test data set. The test data set consists of 746 objects with each 17 features assigned to twelve classes. The dedicated training data set used to learn the membership function comprises 17 images per class, hence 204 images. This represents a typical situation occurring in classification applications, where the training data set from which a robust classifier is to be derived is very small. For details about the data set we refer to [6]. The subsequent classification is executed with different aggregation operators by using the classifier framework presented in [6]. Here, the incorporated aggregation operators are Yager's family of*Ordered Weighted Averaging (OWA)* [14] and Larsen's family of *Andness-directed Importance Weighting Averaging (AIWA)* [15] operators (applied unweighted here)—which both can be adjusted in their andness degree—and additionally MFPC's original geometric mean (GM). Due to space limitations, we refer to [14] and [15] for the definition of OWA and AIWA operators. As a reference, the data set is also classified using a *Support Vector Machine (SVM)* with a Gaussian radial basis function (RBF). Since SVMs are capable of distinguishing between only two classes, the classification procedure is adjusted to pairwise (or one-against-one) classification according to [16]. Our benchmarking measure is the classification rate $r_+ = \frac{n_+}{N}$, where n_+ is the number of correctly classified objects and N the total number of objects that were evaluated. The best classification rates at a given aggregation operator's andness ρ_g are summarised in the following Table 1, where the best classification rate per group is printed bold.

Table 1. "OCR" classification rates r_+ for each aggregation operator at andness degrees ρ_g with regard to membership function parameters D and p_{C_e}

	Aggregation Operator	μ_{PMFPC}		μ_{MFPC}							
				$D = 2$		$D = 4$		$D = 8$		$D = 16$	
ρ_g		p_{C_e}	r_+	p_{C_e}	r_+	p_{C_e}	r_+	p_{C_e}	r_+	p_{C_e}	r_+
0.5000	AIWA	**0.255**	**93.70 %**	0.370	84.58 %	0.355	87.67 %	0.310	92.36 %	0.290	92.90 %
	OWA	**0.255**	**93.70 %**	0.370	84.58 %	0.355	87.67 %	0.310	92.36 %	0.290	92.90 %
0.6000	AIWA	**0.255**	**93.16 %**	0.175	87.13 %	0.205	91.02 %	0.225	92.36 %	0.255	92.23 %
	OWA	**0.255**	**93.57 %**	0.355	84.58 %	0.365	88.47 %	0.320	92.63 %	0.275	92.76 %
0.6368	GM	**0.950**	**84.45 %**	0.155	81.77 %	0.445	82.17 %	0.755	82.44 %	1.000	82.44 %
	AIWA	**0.245**	**91.42 %**	0.135	85.52 %	0.185	90.08 %	0.270	89.81 %	0.315	89.95 %
	OWA	**0.255**	**93.57 %**	0.355	84.72 %	0.355	88.74 %	0.305	92.63 %	0.275	92.76 %
0.7000	AIWA	**1.000**	**83.65 %**	0.420	82.71 %	0.790	82.57 %	0.990	82.31 %	1.000	79.22 %
	OWA	**0.280**	**93.57 %**	0.280	84.85 %	0.310	89.01 %	0.315	92.76 %	0.275	92.63 %

The best classification rates for the "OCR" data set are achieved when the PMFPC membership function is incorporated, which are more than 11 % better than the best incorporating μ_{MFPC}. The Support Vector Machine achieved a best classification rate of $r_+ = 95.04\,\%$ by parameterising its RBF kernel with $\sigma = 5.640$, which is 1.34 % or 10 objects better than the best PMFPC approach.

5 Conclusion and Outlook

Based on the MFPC membership function, we developed and presented a probabilistic parameterisation method, which automatically learns the membership functions based on a given set of training data. This method yields membership functions which outperform any approach using μ_{MFPC} as a fuzzy classifier's membership functions and provides a performance similar to a Support Vector Machine for the evaluated sample data set. Nevertheless, the presented approach is not intended to serve as a SVM substitute, but to show its possible performance compared to a state-of-the-art classification technique while providing robust results for small training data sets and preserving real-time demands as well as hardware-implementability. All results obtained must be seen in the scope of the test case, general statements cannot be derived. Still more data sets need to be classified to see if the trends hold.

Acknowledgement. This work was supported and financed by the German Federal Ministry of Education and Research, Grant-No. 17N2708.

References

1. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy sets and systems 1, 3–28 (1978)
2. Bocklisch, S.F.: Prozeßanalyse mit unscharfen Verfahren. Verlag Technik, Berlin (1987)
3. Lohweg, V.: Ein Beitrag zur effektiven Implementierung adaptiver Spektraltransformationen in applikationsspezifische integrierte Schaltkreise. PhD thesis, Technische Universität Chemnitz (2003)
4. Lohweg, V., Diederichs, C., Müller, D.: Algorithms for hardware-based pattern recognition. EURASIP Journal on Applied Signal Processing 2004(12), 1912–1920 (2004)
5. Niederhöfer, M., Lohweg, V.: Application-based approach for automatic texture defect recognition on synthetic surfaces. In: IEEE International Conference on Emerging Technologies and Factory Automation, vol. (19), pp. 229–232 (2008)
6. Mönks, U., Lohweg, V., Larsen, H.L.: Aggregation Operator Based Fuzzy Pattern Classifier Design. In: Lohweg, V., Niggemann, O. (eds.) Lemgo Series on Industrial Information Technology, vol. 3. IT – Institut Industrial IT, Lemgo (2009)
7. Lohweg, V., Li, R., Türke, T., Willeke, H., Schaede, J.: FPGA-based Multi-sensor Real time Machine Vision for Banknote Printing. In: 21st annual Symposium on IS&T/SPIE Electronc Imaging, San Jose, California USA (2009)
8. Rodner, E., Denzler, J.: Learning with Few Examples by Transferring Feature Relevance. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 252–261. Springer, Heidelberg (2009)
9. Drobics, M., Bodenhofer, U., Peter Klement, E.: FS-FOIL: an inductive learning method for extracting interpretable fuzzy descriptions. International Journal of Approximate Reasoning 32(2-3), 131–152 (2003)
10. Eichhorn, K.: Entwurf und Anwendung von ASICs für musterbasierte Fuzzy-Klassifikationsverfahren am Beispiel eines Schwingungsüberwachungssystems. PhD thesis, Technische Universität Chemnitz (2000)
11. Salicone, S.: Measurement Uncertainty: An Approach via the Mathematical Theory of Evidence. Springer Series in Reliability Engineering. Springer Science+Business Media LLC, Boston (2007)
12. Polyanin, A.D., Manzhirov, A.V.: Handbook of mathematics for engineers and scienctists. Chapman & Hall/CRC, Boca Raton (2007)
13. Dujmović, J.J., Larsen, H.L.: Generalized conjunction/disjunction. International Journal of Approximate Reasoning 46(3), 423–446 (2007)
14. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decisionmaking. IEEE Transactions on Systems, Man and Cybernetics 18(1), 183–190 (1988)
15. Larsen, H.L.: Efficient Andness-Directed Importance Weighted Averaging Operators. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 11(suppl. 1), 67–82 (2003)
16. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge (2001)