Handwritten Brazilian Month Recognition: An Analysis of Two NN Architectures and a Rejection Mechanism

MARCELO N. KAPP[1], CINTHIA O. DE A. FREITAS [1], ROBERT SABOURIN[2]

[1]Pontifícia Universidade Católica do Paraná (PUCPR) – Rua: Imaculada Conceição, 1155 – Prado Velho – 80215-901 – Curitiba – PR – Brazil {mnk,cinthia, nievola}@ppgia.pucpr.br
[2] École de Technologie Supérieure (ETS) – 1100, Rue Notre Dame – Ouest – H3C 1K3 – Montreal (QC) – Canada robert.sabourin@.etsmtl.ca

Abstract

This paper evaluates the use of the conventional architecture feedforward MLP (multiple layer perceptron) and class-modular for the handwriting recognition (HWR) and it also compares the results obtained with previous works in terms of recognition rate. This work presents a feature set in full detail to work with HWR. The experiments showed that the class-modular architecture is better than conventional architecture. The obtained average recognition rates were 77.08% using the conventional architecture and 81.75% using the class-modular. This paper also describes a performance study in which a rejection mechanism with multiple thresholds is evaluated for both conventional and class-modular architectures. The multiple thresholds idea is based on the use of N class-related reject thresholds (CRTs). The results indicate that this rejection mechanism can be used appropriately in both architectures. The experimental results are 86.38% and 91.52% using a handwritten months word database

1. Introduction

The main objective of this work is to evaluate the performance of a Conventional architecture feedforward MLP (multiple layer perceptron) in relation to Class-Modular architecture for the recognition of the handwritten names for the months of the year in Brazilian Portuguese language. This is an important task since it constitutes a sub-problem of bank check date recognition. Although this study deals with a limited lexicon of 12 classes, there are classes that share a common sub-string, which can affect the overall system performance: Jan**eiro**, Fever**eiro**, **Ma**rço, Abril, **Ma**i**o**, **Ju**n**ho**, **Ju**l**ho**, Agosto, Se**tembro**, Outu**bro**, Nov**embro** and Dez**embro** [1].

In general, handwriting recognition generates high-dimensional problems [2]. This work also suggests a simple feature set that makes possible the recognition in relatively reduced dimensions. The power of Artificial Neural Networks (ANNs) resides in its capacity to generate an area of decision of any form. However, different performances can be obtained with the conventional and modular architectures. Modularity is an essential concept, which should be used appropriately in the design of systems for diverse application areas. Since K classes are involved in the classification module, we can naturally think of the classes as a target of modularity. It leads us directly to the *class modularity* concept [2]. In the class-modular concept, each class should be managed independently of the other classes, at least conceptually [2]. In this work the conventional and class-modular feedforward neural network architectures are evaluated based on a feature set and applying global techniques for the extraction of patterns.

Usually, recognition systems apply a global decision module which decides either to accept the recognition result or reject it. In classification, a pattern is considered ambiguous if it cannot be reliably assigned to a class, whereas a pattern assigned low confidence for all hypothesized classes can be treated as an outlier.

The purpose of a rejection mechanism is to minimize the number of recognition errors for a given number of rejects. A simple rejection scheme involves the rejection of an image with a global probability lower than a determined threshold, as denoted by Chow's rule $y_i < T$ [3]. In this paper, we investigate the effects of estimate errors on Chow's rule and CRTs based on multiple reject thresholds related to the data classes, as [4], however we also investigate the effects in the class-modular architecture. The reported experimental results show that such class-related reject thresholds provide a better error-reject trade-off than that in Chow's rule.

This paper is organized as follows. The Section 2 describes the feature set extracted from the word images. Section 3 and Section 4 introduce respectively the Conventional and Class-Modular architectures. In Section 5 the experimental results are provided with some analyses and discussions. Section 6 presents the rejection mechanism applied and the obtained results. Section 7 presents the concluding remarks and future work.

Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)
0-7695-2187-8/04 $20.00 © 2004 IEEE

2. Features Extraction

The most of the pattern recognition studies and more specifically of the handwritten one have as one of its relevant points the feature set selection which must represent and discriminate the different found shapes.

In this work, perceptual features [5] and characteristics based on concavities / convexities and another, were explored for the recognition of handwritten names of the months of the year in Brazilian Portuguese language. Basically, they are numbers of occurrences of such features. However, only these discrete primitives do not make the recognition system more robust [6]. Therefore, it was added to the features set a zoning mechanism during the extraction of the primitives.

The zoning is used only in two areas separated by the center of gravity of the word: *left-area* and *right-area*, as shown in Figure 1. It was chosen because in each midfield the occurrence of some features gives more useful information for the pattern classifier.

The ascending and descending zones are computed taking into account the Upper (UL) and Lower (LL) lines. UL and LL are based on maximum horizontal projection histogram of black-white transitions, establishing the central line (CL), as presented in Figure 2.

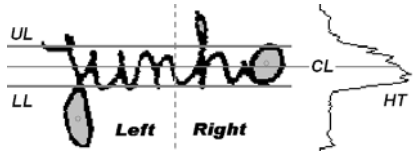


Figure 1: Example of zoning mechanism and areas detection

The feature set can be described as following:

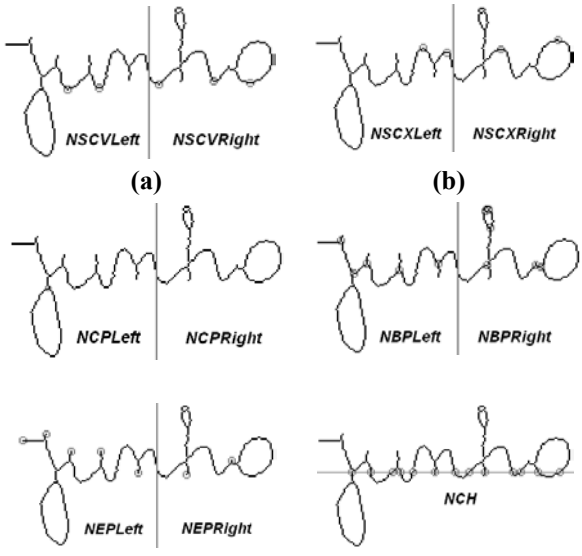
- Number of loops on the *left/right-areas* (NLL=2 and NLR=3), Figure 1;
- Number of concave semicircles on the *left/right-areas* (NSCVL=3 and NSCVR=5), Figure 3-a;
- Number of convex semicircles on *left/right-areas* (NSCXL=3 and NSCXR=3), Figure 3-b. The concavities and convexities are only extracted in the tuned words. The concave and convex points are obtained by mathematical morphology;
- Number of crossing-points on the *left/right-areas* (NCPL=1 and NCPR=1), Figure 3-c;
- Number of branch-points on the *left/right-areas* (NBPL=3 and NBPR=6), Figure 3-d;
- Number of end-points on the *left/right-areas* (NEPL=3 and NEPR=1), Figure 3-e;
- Number of crossings between the stroke and the horizontal axis (NCH), Figure 3-f;
- Number of ascenders on the *left/right-areas* (NAL=0 and NAR=0);
- Number of descenders on the *left/right-areas* (NDL=1 and NDR=0);
- Proportion of black pixels in relation to the white one (NPP=0.955324), Figure 3-g. The pixels proportion is part of the surface in relation to the context of the word (NPP). A bounded box is used and the proportion can be obtained by the Equation (1) computed inside the bounded box, as follows:

$$prop = (tp - tpp) / tp \qquad (1)$$

  where *tp* is the total of pixels inside the bounded box and *tpp* is the total of black pixels of the word stroke,

- Number of vertical lines (NVL=7), Figure 3-h,
- Number of horizontal lines (NHL=0),
- Number of ascenders with loop on the *left/right-areas* (NALL=0 and NALR=0),
- Number of descenders with loop on the *left/right-areas* (NDLL=1 and NDLR=0).

These 14 features are extracted from each word in order to generate a feature vector of 24 dimensions. When a feature is not found in the word, a small value is assumed, for our case, 0.001.



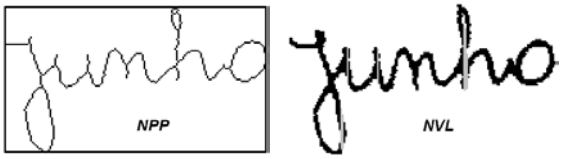


Figure 3: Feature extraction: a) concave semicircles, b)convex semicircles, c) crossing-points, d) branch-points, e)end-points, f) NCH, g)NPP, h)vertical lines

3. Conventional Architecture

The MLP has been used extensively in implementing the *K*-classification module for the word recognition. One of distinct properties of the conventional MLP architecture is that all the *K* classes share one large network [2], as shown in Figure 4. The essential task in designing a character recognition system is to choose a feature type with a good discriminative power and the network should divide the *K* class regions well in the chosen feature space.

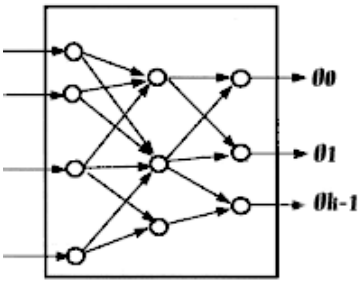


Figure 4: Conventional architecture where *K* classes are intermingled [2]

However, determining the optimal decision boundaries for the *K*-classification module for word recognition in a high-dimensional feature space is very complex, and can seriously limit the recognition performance of the character recognition system using MLP [2,7]. Particularly, when the training set is not large enough compared with the classifier size (i.e., the number of free parameters in the classifier), a problem occurs in convergence [7].

4. Class-modular MLP

A single task is decomposed into multiple subtasks and each subtask is allocated to an expert network. In this paper, as well as in [2], in the class-modular classification, the *K*-classification problem is decomposed into *K* 2-classification subproblems, each for one of the *K* classes. A 2-classification subproblem is solved by the 2-classifier specifically designed for the corresponding class. The 2-classifier is only responsible for one specific class and discriminates that class from the other *K-1* classes. In the class-modular framework, *K* 2-classifiers solve the original *K*-classification problem cooperatively and the class decision module integrates the outputs from the *K* 2-classifiers.

In Figure 5, we can see the MLP architecture for 2-classifier. The modular MLP classifier consists of *K* sub-networks, M_i for $0 \le i \le K-1$, each responsible for one of the *K* classes. The architecture for the entire network constructed by *K* sub-networks is shown in Figure 6.

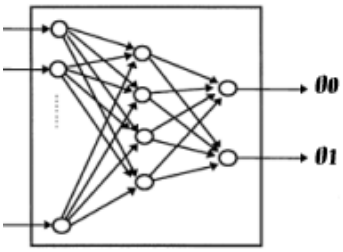


Figure 5: Class-modular architecture: sub-network [2]

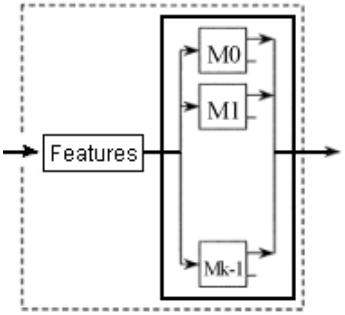


Figure 6: Architecture for the class-modular: whole network with M modules [2]

5. Experimental Results

This section describes the database and presents the results obtained with the conventional and class-modular MLP architectures.

5.1 Database

The database used is composed by names of the months of the year and was collected by UFPB (Federal University of Campina Grande-Paraíba-Brazil), for more details see [1]. In total there are 6000 word images, with 500 of each class. All the images are already preprocessed, i.e., the baseline skew and slant were corrected, reducing the writing variability (different writing styles and particular writing characteristics). For the experiments, the database was randomly divided in three data sets: Training

set (60%), Validation set (20%) and Test set (20%). Figure 7 shows sample images from the database.

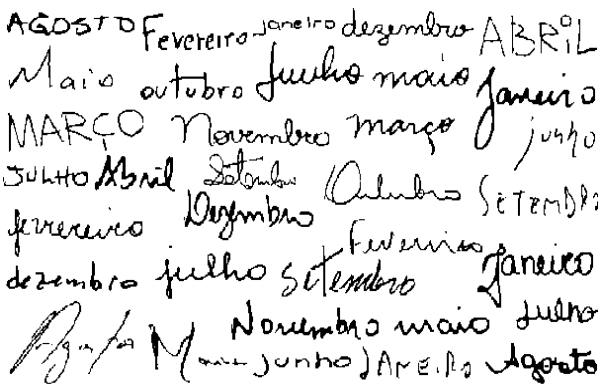


Figure 7: Sample images from the database

5.2 Conventional Architecture Results

Conventional MLP is composed by 24 nodes in the input layer, 45 nodes in the hidden layer and an output layer with 12 nodes. Validation sets were employed in order to avoid over-training. The stop criterion is the increase of the error read in the validation set.

All the classes are trained together. The class that presents the maximum output value is the class considered as recognized. The recognition rate obtained for the conventional architecture is 77,08%. The confusion matrix for the test set is shown in Table 1.

5.3 Class-modular MLP Results

In class-modular MLP, each of K 2-classifier is trained independently of the other classes using the training and validation set. The backpropagation algorithm was used in each of 2-classifiers in the same way as in conventional MLP. To train 2-classifier for each word class (n = 12), we re-organize the samples in the original training and validation set into n-two groups, *Z0* and *Z1* such that *Z0* has the samples from current class and *Z1* all the other ones, taking account the *a priori* probability for each class.

To recognize the input word patterns, the class decision module takes only the values of O_0 and uses the simple winner-take-all scheme to determine the final class. A conventional network sees each of the training instances once per epoch. However, in the case of modular network, each subnetwork sees each training instance once per epoch, so the whole network sees each sample K times per epoch [2]. The recognition rate obtained for the class-modular architecture was 81,75%. The confusion matrix for this experiment is presented in Table 2.

Table 1: Confusion Matrix for conventional architecture

Month	J	F	M	A	M	J	J	A	S	O	N	D
Jan	**77**	5	4	0	1	1	1	2	1	2	5	1
Fev	6	**87**	0	0	0	0	0	0	2	4	0	1
Mar	5	1	**74**	2	12	2	2	1	0	1	0	0
Abr	0	1	5	**84**	4	3	2	1	0	0	0	0
Mai	1	2	9	5	**77**	0	2	1	1	2	1	0
Jun	2	2	1	0	4	**77**	10	0	1	3	0	0
Jul	1	1	2	3	4	10	**73**	3	0	3	0	0
Ago	6	2	5	4	0	1	2	**73**	0	0	2	5
Set	2	8	0	1	0	1	0	0	**71**	7	6	4
Out	3	3	3	0	0	2	1	0	9	**76**	3	0
Nov	0	2	6	0	0	1	0	0	3	1	**82**	5
Dez	4	6	0	0	1	1	1	2	4	1	6	**74**

Table 2: Confusion Matrix for class-modular architecture

Month	J	F	M	A	M	J	J	A	S	O	N	D
Jan	**83**	8	2	0	0	2	0	0	0	2	1	2
Fev	5	**83**	1	1	1	0	0	1	3	1	2	2
Mar	3	3	**75**	5	10	0	0	0	0	2	2	0
Abr	1	1	1	**93**	2	0	2	0	0	0	0	0
Mai	1	0	10	5	**80**	2	1	0	0	0	1	0
Jun	1	3	0	0	5	**84**	4	0	1	2	0	0
Jul	1	0	0	6	4	9	**76**	0	0	3	1	0
Ago	2	4	2	3	0	3	0	**78**	0	3	3	2
Set	1	10	0	0	0	0	0	0	**73**	6	8	2
Out	4	4	2	0	0	0	0	0	4	**85**	1	0
Nov	3	2	0	0	0	0	0	0	4	1	**89**	1
Dez	3	5	0	0	0	2	1	1	0	0	6	**82**

5. 4 Discussions

Table 3 summarizes the result obtained in this work and in some other studies [1]. Observe that ANNs in general obtained best results than HMM (Hidden Markov Models), when a similar feature set is applied. The Conventional network obtained better recognition rate than Directional Features (DF)/ANN. The class-modular network obtained recognition rate similar to the use of the Perceptual Features (PF)/ANN. More than that, each module could yet be optimized, aiming to better rates. Two concluding remarks can be made based on experimental results, as following:

- The class-modular network was superior in terms of the convergence over conventional network (according to the monitoring of the MSE – mean square error); and
- The class-modular network was also superior in terms of recognition capability regarding the conventional network.

Table 3: Comparison of word recognition results

Set	Recognition
HMM [1]	75.90 %
Conventional Architecture	**77.08 %**
Class-modular MLP	**81.75 %**

6. Rejection Mechanism

An N-class classifier is aimed at subdividing the feature space into N decision regions D_i, i=0,.., N-1, such that the patterns of class w_i belong to region D_i. According to statistical pattern recognition theory, such decision regions are defined so as to maximize the probability of correct recognition, commonly referred to as classifier accuracy:

$$Accuracy = P(correct) = \sum_{i=0}^{N-1} \int_{D_i} p(x \mid w_i) P(w_i) dx \quad \textbf{(1)}$$

Consequently, to minimize classifier error probability:

$$P(error) = \sum_{i=0}^{N-1} \int_{D_i} \sum_{\substack{j=0 \\ j \neq i}}^{N-1} p(x \mid w_j) P(w_i) dx \quad \textbf{(2)}$$

To this end, the so-called Bayes decision rule assigns each pattern x to the class for which the *a posteriori* probability $P(w_i|x)$ is at its maximum. An error probability lower than that provided by the above Bayes rule can be obtained using the so-called "reject" option [4]. Namely, the patterns that are the most likely to be misclassified are rejected (i.e. they are not classified). Therefore, a trade-off between error and reject is mandatory. The formulation of the best error-reject trade-off and the related optimal reject rule was given by Chow [3]. A careful analysis of Chow's work allows us to point out that his reject rule provides the optimal error-reject trade-off only if the *a posteriori* probabilities are known exactly. Therefore, in Fumera et al [4], the authors suggest the use of multiple reject thresholds to obtain the optimal decision and reject regions, even if the *a posteriori* probabilities are affected by errors. It is easy to see that such thresholds applied to the estimated probabilities make it possible to obtain both the optimal decision regions and the rejection region. This experiment therefore suggests that the use of N class-related reject thresholds (CRTs) can provide a better error-reject trade-off than Chow's rule [4], and also in class-modular architecture. In particular, under the assumption that the *a posteriori* probabilities are affected by significant errors, the authors have proved in [4] that, for any reject rate R, such values of the CRTs $T_0,..., T_{N-1}$ exist such that the accuracy of the corresponding classifier A($T_0,..., T_{N-1}$) is equal to or higher than the accuracy A(T) provided by Chow's rule, see (Equation 7):

$$\forall R \exists T_0, T_1, ..., T_{N-1} : A(T_0, T_1, ..., T_{N-1}) \geq A(T) \quad \textbf{(3)}$$

The authors therefore proposed in [4] the following reject rule, named the CRT rule, for a classification task with N data classes which are characterized by estimated a posteriori probabilities $\hat{P}(w_i \mid x), i = 0,..., N-1$. A pattern x is rejected if:

$$\max_{k=0,...,N-1} \hat{P}(w_k \mid x) = \hat{P}(w_i \mid x) < T_i \quad \textbf{(4)}$$

While it is accepted and assigned to class w_i if:

$$\max_{k=0,...,N-1} \hat{P}(w_k \mid x) = \hat{P}(w_i \mid x) \geq T_i \quad \textbf{(5)}$$

The CRTs take on values in the range [0,1]. It is worth noting that, by analogy with Chow's rule, the values of the CRTs must be estimated according to the classification task at hand in real applications. In our experiments, such as in [4]. Accordingly, the CRT values were estimated by solving the following constrained maximization problem (Equation 6):

$$\begin{cases} \max_{T0,....,TN-1} A(T_0,...., T_{N-1}) \\ E(T_0,..., T_{N-1}) \leq E_{MIN} \end{cases} \quad \textbf{(6)}$$

It is worth noting that, according to (Equation (3)), for any given E_{MIN}, the CRT values obtained as solutions of the above maximization problem provide an accuracy equal to or higher than that in Chow's rule. Therefore, [4] takes on a finite number of values in the range [0, 1] and (Equation (5)) represents a constrained maximization problem the "target" and "constraint" functions of which are discrete valued functions of continuous variables. Our algorithm takes into account that $E(T_0,...,T_{N-1})$ is an increasing function of the variables $T_0,\ldots,T_{N-1}$ (i.e. the number of rejected patterns cannot decrease for increasing CRT values) and also assumes that $A(T_0,..., T_{N-1})$ is an increasing function $T_0,... , T_{N-1}$. All rejection thresholds used for the experiments on the test set (Table 3) were estimated on the validation set.

We have observed in Tables 4 and 5 that, as well as in [4], all thresholds obtained through CRTs was better than those obtained with the Chow's rule for a conventional architecture MLP. Moreover, we have observed also that a class-modular architecture was superior to a conventional one, with larger recognition rates in smaller rejection rates and in fixed error rates, see Table 5. The explanations for these facts are that, part of this, due to a best mapping

among features space and classes provided by the architecture class-modular before even of the rejection process, and mainly the idea of multiple thresholds, obtained in local way in each module for each class and not global as in the Table 4. The superiority also extends to a test set unknown by the architectures (see Table 6). However, the error rates differ of those observed with the validation set due to the intrinsic variability of writing styles (cursive pure, printed, mixed and others) and a different frequency of occurrence of these writing styles in both data sets.

Table 4: Chow's rule and validation set

Error Rates	Conventional			Class-Modular		
	Rec.	Rej.	Rel.	Rec.	Rej.	Rel.
1%	21.34	77.66	95.52	23.00	76.00	95.83
2%	26.75	71.25	93.04	29.75	68.25	93.70
5%	40.25	54.75	88.95	46.08	48.92	90.21

Rec.=Recognition Rate, Rej.=Rejection Rate and Rel.=Reliability Rate

Table 5: Multiple thresholds and validation set

Error Rates	Conventional			Class-Modular		
	Rec.	Rej.	Rel.	Rec.	Rej.	Rel.
1%	58.75	40.25	98.16	61.33	37.67	98.29
2%	66.08	31.92	96.81	68.33	29.67	96.97
5%	72.42	22.58	93.51	75.50	19.50	93.73

Table 6: Multiple thresholds and test set

Rates	Conventional			Class-Modular		
	1%	2%	5%	1%	2%	5%
Rec.	57.17	67.00	75.42	68.33	74.17	79.75
Rej.	34.00	20.42	7.33	25.33	17.08	6.92
Error	8.83	12.58	17.25	6.33	8.75	13.33
Rel.	86.62	84.19	81.38	91.52	89.45	85.68

7. Conclusions

The results indicate that this research is quite promising and prove to be worthy of further investigations of the class modularity paradigm. A consideration must be made about large-set classification in order to test the effect of the number of classes on the recognition capability (for example: legal amounts). We proposed and implemented a new feature set with smaller dimension than presented in [1]. Then it generates less parameter to be estimated in the ANN, decreasing the complexity computation without loss of recognition performance. We also observed that the class-modular network was superior in terms of convergence and recognition capability over the conventional network, such as [2].

The conventional architecture has a *rigid* structure composed of an unstructured black box in which all the *K* classes are altogether intermingled. The modules cannot be modified or optimized locally for each class. However, the disadvantages of the class-modular architecture are firstly the reorganizations of training, validation and test set to assist each class as described in the Section 5.3, and the training of *K* networks for the classes of the problem.

The obtained results motivate the continuity of the system development considering a rejection mechanism. Other future work is study a feature set for each one (global and modular architecture) based on dependent-class feature subsets. Based on an analysis of Tables 4, 5 and 6, we can say that a better error-reject trade-off can be obtained with the rejection rule proposed by [4] mainly because it represents a local search of each class and not a global search as in [3]. Accordingly, this rejection mechanism also behaves better in the class-modular architecture.

8. References

[1] J. J. Oliveira Jr., J. M. de C. Carvalho, C. O. de A. Freitas, R. Sabourin, "Evaluating NN and HMM classifiers for handwritten word recognition", *15 th Brazilian Symposium on Computer Graphics and Image Processing* (2002), 210--217.

[2] I-S. Oh, C. Y. Suen, "A class-modular feedforward neural network for handwriting recognition", *Pattern Recognition* 35 (2002), 229--244.

[3] C.K. Chow, "On optimum error and reject tradeoff", *IEEE Trans. Inform. Theory*, 16(1):41-46, 1970.

[4] G. Fumera, F. Roli, G. Giacinto, "Reject option with multiple thresholds", *Pattern Recognition* 33 (2000), 2099-2101.

[5] S. Madhvanath, V. Govindaraju, "The Role of Holistic Paradigms in Handwritten Word Recognition", *IEEE Trans. on PAMI* 23 (2001), 149--164.

[6] O. D. Trier, A. K. Jain, T. Taxt, "Feature extraction methods for character recognition-a survey", *Pattern Recognition* 29 (1996), 641-662.

[7] A. K. Jain, R. P. W. Duin, J. Mao, "Statistical Pattern Recognition: A Review", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), 4--37.