Prototype Selection for Handwritten Connected Digits Classification

Cristiano de Santana Pereira[1] and George D. C. Cavalcanti[2]

[1]Federal Institute of Pernambuco - Department of Electro-electronics and Systems
[2]Center of Informatics, Federal University of Pernambuco
www.cin.ufpe.br/∼viisar
Recife, PE, Brazil
{cdsp,gdcc}@cin.ufpe.br

Abstract

After the handwritten segmentation process, it is common to have connected digits. This is due to the great size and shape digit variations. In addition, the acquisition and the binarization processes can add noise to the images. These under segmented images, when given as input to classifiers which are specialists to deal with digits separately, should lead to errors. Aiming to detect the handwritten connected digits, it is herein introduced a hybrid system architecture to be used as a segmentation pos-processing task. The proposed system is based on a prototype selection scheme that combines self-generating prototypes and Gaussian mixtures. Besides, this work presents a set of features for the proposed problem. A real-world database of handwritten digits was used to validate the new approach. The results obtained in the experimental study showed that the hybrid strategy achieved promising accuracy rates.

1. Introduction

Segmentation is an important step in the task of handwritten digits recognition. Failures in this step lead to errors in the classification process. These failures are caused by over- or under-segmentation. Over-segmentation occurs when a digit is divided in two or more parts. In opposition, the under-segmentation occurs when the segmentation algorithm is not able to isolate the digits presented in an image, resulting in images with connected digits.

Some previous works using different approaches have shown the relevance of this topic for the handwritten character recognition. For instance, Alhajj and Elnagar [1] employed multiagents to detect the deepest-top valley and the highest-bottom hill to find the actual cut-point of connected digits. Wang et al. [7] proposed modifications on the grid sizes and weighted score in the isolated character recognizer to support the holistic recognition of touching digits pairs and touching character pairs.

The main purpose of this work is to build a system to detect handwritten connected digits based on prototype selection. As a secondary contribution, a set of features to deal with this detection problem is suggested. Prototype selection schemes has been used in many applications when a large amount of data is available. These techniques can be classified as *selective* or *creative* [5]. In the former one, the result set of prototypes are totally formed by data points from the original dataset. In *creative* schemes, new prototypes are created during the reduction process by combining data points.

The next section shows the system architecture and describes in details the main modules. Section 3 describes the database, the methodology and the results obtained during the experimental study. In Section 4 the conclusions about this work are presented.

2. System Architecture

The proposed system architecture is showed in Figure 1. The first step of this process is the feature extraction. In this step, numeric features are calculated from black-and-white images of the handwritten digits. Details about the feature extraction algorithms are explained in Section 2.1. The second step is the prototype initialization. The Self-generating prototypes (SGP) [4] algorithm is used for this task. In Section 2.2, it is shown and explained each step of the SGP algorithm. As results, the SGP gives an initial set of prototypes that is used as input to the Soft nearest prototype classifier (SNPC) [6]. SNPC will adjust the prototypes location aiming a better definition of the classes boundaries. The result set of prototypes from this last step will be used as the trained set using the SNPC decision rules. In Section 2.3, it is explained the mathematical basis of the SNPC and its training and classification rules.

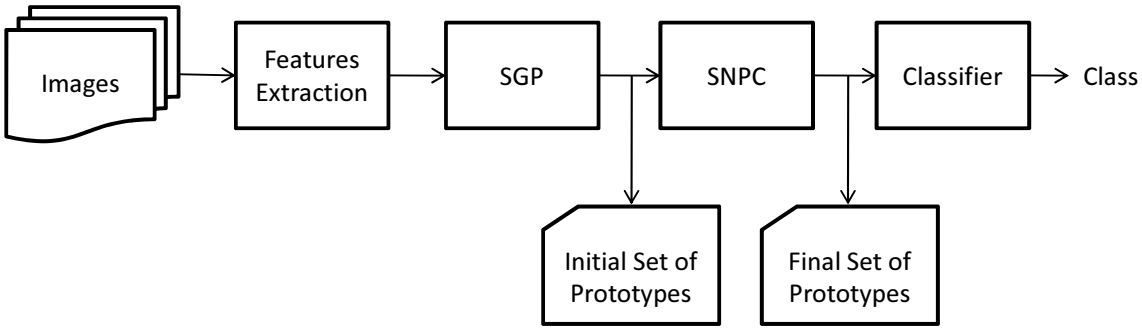


Figure 1. System architecture

Table 1. Features

Measures	
$C_1 \dots C_7$	Counting of Black-to-white transitions
C_8	Number of lines
C_9	Number of columns
C_{10}	C_9/C_8
C_{11}	$Peak_1/C_9$
C_{12}	$Peak_2/C_9$

2.1. Feature Extraction

A set of 12 features developed to deal with connected digits is here introduced. A resume of this set of features can be viewed in Table 1. The first seven features $C_1, C_2, \dots, C_7$ aim to count the number of black-to-white transitions. The image is divided into seven horizontal slices. For each slice i, C_i represents the total number of black-to-white transitions. In other words, each slice contains a set of lines, C_i represents the sum of black-to-white transitions in each one of the lines in the i-th slice. The slices are equally spaced and the distance in lines between them is specified by:

$$\text{DIST} = \left\lfloor \frac{\text{L}}{(\text{NS}+1)} \right\rfloor \tag{1}$$

where L is number of image lines and NS is the number of slices. In this work, the number of slices was fixed to 7 after preliminary experiments. The features C_8 and C_9 are the number of lines and columns, respectively. The ratio between the height and the width of the image is given by $C_{10} = C_9/C_8$. The two last features represent the two highest peaks in the horizontal projection of the image. Aiming to avoid noise peaks, only the projection points having neighbors with at least 80% of the peak value is taken into account.

2.2. SGP

Like other prototype-based machine learning schemes, SNPC depends on the correct choice of the number of prototypes per class and their initial locations. It is often difficult to find the optimal values for these parameters and the algorithm must run a number of times using different sets of parameters to do so. This takes a long time and increases the computational overhead. The Self-Generating Prototype (SGP) method attempts to overcome these problems. The main advantage of this method is that the number of prototypes and their locations are obtained during the training process without much human intervention. Another advantage is the simplicity of its prototype selection strategy. The data points that have the same class label are grouped and the mean of each group represents the initial prototype set, resulting in one prototype per group. During the training process, the following operations are performed: groups are divided; data points are changed from one group to another; some groups are merged; and some groups are removed after a pruning step. A more detailed explanation about the SGP can be found in Fayed et al. [4].

2.3. SNPC

The SNPC is a soft training strategy for the nearest prototype classifier (NPC). SNPC introduces fuzzy assignment probabilities to the data points in relation to the prototypes. Thus, for each data point, the degree of assignment of the point to all prototypes in the problem is defined. During the learning process, the prototypes are adjusted in order to maximize the correct assignments and minimize the classification error.

In order to guide the learning and evaluate the generalization performance, a cost function (2) has been introduced.

$$E = \frac{1}{N} \sum_{k=1}^{N} \sum_{\{j:c_j \neq y_k\}}^{M} P(j|x_k) \tag{2}$$

where N is the number of data points, M is the number of prototypes, x_k is a sample point, y_k is the true class of x_k and c_j is the j-prototype class.

In SNPC, the a posteriori probability $P(j|x_k)$ is defined as:

$$P(j|x_k) = \frac{exp(-d(x_k, \theta_j))}{\sum_{m=1}^{M} exp(-d(x_k, \theta_m))} \quad (3)$$

where $d(x_k, \theta_j)$ is the distance from x_k to the prototype θ_j. Equation (3) represents the fuzzy assignment probabilities of the model. Note that (3) is a continuous function and a gradient descendent-based optimization procedure is therefore possible [2].

The individual cost of a sample is the sum of its assignment probabilities to all prototypes of the incorrect classes and the total classifier cost is the sum of all individual costs. From Equation (2), the individual cost ls_k for each point is obtained as follows:

$$ls_k = \sum_{\{j:c_j \neq y_k\}}^{M} P(j|x_k) \quad (4)$$

Prototype positions are changed following equation (5).

$$\theta_l(t+1) = \theta_l(t) - \alpha(t)\frac{\partial ls_t}{\partial \theta_l} \quad (5)$$

The learning rule (6) was obtained from (5) and (2) and a formal proof can be viewed in [6].

$$\begin{aligned} \theta_l(t+1) &= \theta_l(t) - \alpha(t)\Delta\theta_l(t) \quad (6) \\ \Delta\theta_l(t) &= \begin{cases} P(l|x_t) ls_t \frac{\partial d(x,\theta_l)}{\partial \theta_l} & c = y \\ -P(l|x_t)(1 - ls_t)\frac{\partial d(x,\theta_l)}{\partial \theta_l} & c \neq y \end{cases} \end{aligned}$$

Once the learning process is finished, a new data point is classified using:

$$c = \underset{c'}{\operatorname{argmax}} \sum_{\{j:c_j = c'\}} P(j|x_k). \quad (7)$$

A D-dimensional Gaussian Mixture Ansatz is assumed to describe the point distribution around the prototypes, thus the probability density is given by:

$$p(x|j) = \frac{1}{(2\pi\sigma^2)^{D/2}} exp\left(-\frac{(x-\theta_j)^2}{2\sigma^2}\right) \quad (8)$$

and the assumed distance measure is:

$$d(x, \theta_j) = \frac{(x-\theta_j)^2}{2\sigma^2} \quad (9)$$

From these assumptions and using (3) and (5), it is possible to obtain a new assignment probability function (10) and new learning rule (11).

$$P(j|x) = \frac{exp\left(-\frac{(x-\theta_j)^2}{2\sigma^2}\right)}{\sum_k exp\left(-\frac{(x-\theta_k)^2}{2\sigma^2}\right)} \quad (10)$$

$$\begin{aligned} \theta_l(t+1) &= \theta_l(t) - \alpha(t)\Delta\theta_l(t) \quad (11) \\ \Delta\theta_l(t) &= \begin{cases} -P(l|x_t) ls_t (x_t - \theta_l) & c = y \\ P(l|x_t)(1 - ls_t)(x_t - \theta_l) & c \neq y \end{cases} \end{aligned}$$

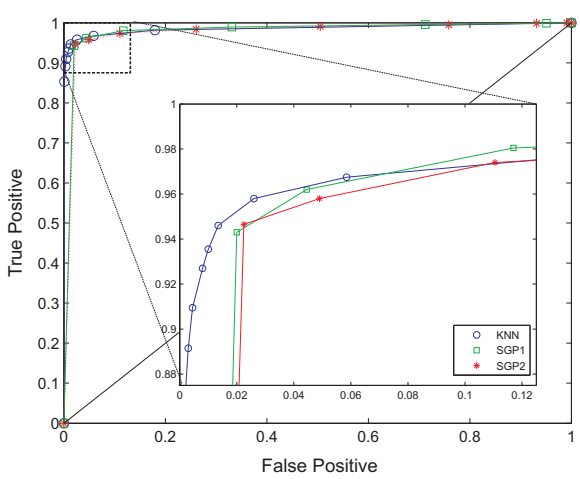
3. Experiments

3.1. Database

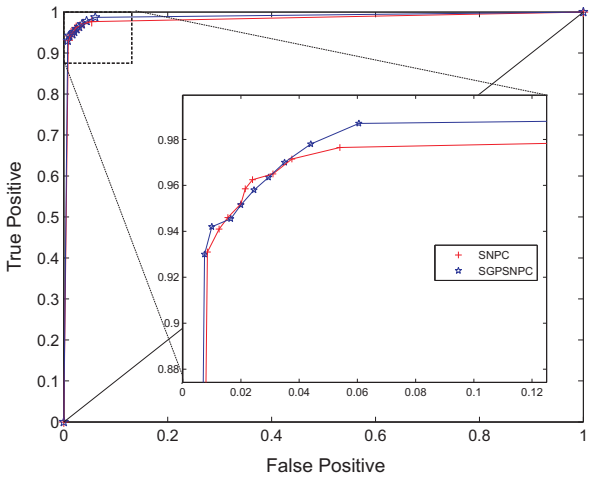
The database used to validate the scheme proposed here is formed by binary images of handwritten digits. This database was built with images acquired from the amount field of bank checks. After, these images were segmented resulting in images of single digits and connected digits. The connected digits images represent examples in which the segmentation algorithm was not able to separate correctly the digits. The class of connected digits can be divided into two subgroups. The former one is composed by images containing two different digits and the second group is totally formed by images that have two digits "0" (zero) connected. It is very common to have this kind of problem in real applications. Sample images of this database is shown in Figure 2.

Figure 2. Handwritten Digits Sample Images

This database has 14,000 samples equally divided in images of single digits and connected digits. It was not applied any normalization process to the images. So, the images present very large variation in size and shape.



(a) KNN, SGP1 and SGP2.



(b) SNPC and SGP+SNPC.

Figure 3. ROC Curves

3.2. Results

The first step of the experimental study was to find the hyperparameters for SGP1, SGP2, SNPC and for the hybrid proposed method. For this, ten-fold cross-validation was performed using a subset of 4,000 images of the database. These images were randomly chosen, but the a priori probabilities of the classes were respected. The sets of parameters that achieved the better accuracy rates were select and used in the whole experimental study. The SGP parameters R_{min} and R_{mis} were tested with values defined between 0 and 1. The SNPC parameters are: the number of prototypes per class ζ, the learning rate α, the standard deviation σ of the Gaussian curves and the threshold of the Window Rule η [6]. ζ was chosen from [3, 15]. α was tested using values from [0.01, 0.20]. σ and η were varied from [0.01, 0.50]. Classical KNN was used for comparison purposes. For the classical KNN, experiments performed over this database pointed to $K = 7$ as the best choice. The hybrid model inherited the SGP hyperparameters. The same ranges used to estimate the SNPC hyperparameters were used to obtain the set of SNPC hyperparameters of the hybrid model. In the case of the hybrid model, the initial set of prototype was supplied by the SGP algorithm. After, the total amount of data was used to get the final results of the experiments. One more time, ten-fold cross-validation was used to estimate the classification error rates. Cross-validation and all the other splitting operations have been stratified.

To evaluate the performance of the proposed model on unbalanced environments, 9 datasets were formed having 4,000 images each one. The unbalanced level is given by the difference between the a priori probability of the classes. As can be seen in the first column of Tables 2, 3 and 4, the unbalanced level changed from 0% to 80%. Thus, the dataset having the greatest level of unbalancing has 90% of single digit images and 10% of connected digits images.

The classification error obtained for each technique is displayed in Table 2; the first column of the table is the unbalanced level. Standard deviations are also displayed. The results in the table are the average of the classification error obtained for the best configurations of each pair ⟨technique,dataset⟩. So, the first line of results shows the classification error rates obtained for the best configuration estimated for the 14,000 images database. The last line and column of the table are the general averages of the techniques and datasets, respectively. Bold numbers represent the lowest classification error and underlined numbers indicate the second lowest classification error.

Note that the hybrid method SGP+SNPC achieved the better accuracy rates in 8 of 9 datasets. For this reason, the hybrid method was the best one in the overall performance evaluation.

Other important evaluation criterium for prototype selection schemes is the number of prototypes. The number of prototypes reflects the reduction capability of the method. The number of prototypes resulting from training process of each method evaluated can be seen in Table 3. SGP2 and the hybrid model obtained the lowest number of prototypes in all datasets.

Another method was used to evaluation the techniques: ROC curves [3]. The Area Under the ROC curve (AUC) is a scalar that has important properties, specially to deal with unbalanced problems. Ten-fold cross-validation was performed and the average of the curve points were illustrated

Table 2. Classification Error Rates

	KNN	SGP1	SGP2	SNPC	SGP+SNPC	$\bar{x}$
0	3.73 ± 0.72	3.98 ± 0.90	3.80 ± 1.02	$\underline{3.30} \pm 0.92$	**2.83** ± 0.73	3.53
10	3.60 ± 1.04	3.47 ± 0.95	3.63 ± 0.80	$\underline{3.12} \pm 0.75$	**2.78** ± 0.90	3.32
20	3.38 ± 0.64	3.50 ± 1.01	3.35 ± 1.11	$\underline{2.93} \pm 0.54$	**2.50** ± 0.67	3.13
30	2.60 ± 1.02	2.73 ± 1.20	2.73 ± 1.20	$\underline{2.10} \pm 0.81$	**2.02** ± 1.01	2.44
40	3.30 ± 0.82	3.48 ± 1.06	3.55 ± 1.09	$\underline{2.80} \pm 0.86$	**2.45** ± 0.77	3.09
50	2.62 ± 1.08	3.02 ± 0.99	2.87 ± 0.95	$\underline{2.45} \pm 0.97$	**2.28** ± 0.82	2.65
60	2.25 ± 0.82	2.02 ± 0.63	2.02 ± 0.63	**1.85** ± 0.68	$\underline{2.08} \pm 0.88$	2.04
70	1.92 ± 0.50	1.90 ± 0.50	1.90 ± 0.50	$\underline{1.80} \pm 0.37$	**1.55** ± 0.47	1.81
80	1.30 ± 0.63	1.45 ± 0.77	1.45 ± 0.77	$\underline{1.15} \pm 0.56$	**1.12** ± 0.57	1.29
$\bar{x}$	2.74	2.84	2.81	2.39	**2.18**	

Table 3. Number of Prototypes

	KNN	SGP1	SGP2	SNPC	SGP+SNPC
0	3600	21	10	14	10
10	3600	20	10	14	10
20	3600	14	6	14	6
30	3600	9	7	14	7
40	3600	8	5	14	5
50	3600	7	4	14	4
60	3600	2	2	14	2
70	3600	2	2	14	2
80	3600	2	2	14	2

Table 4. Area under ROC curve

	k-NN	SGP1	SGP2	SNPC	SGP+SNPC
0	0.9723	0.9754	0.9779	0.9815	**0.9925**

in Figure 3. To turn easier the visualization of the curves, the areas of interest of the images were enlarged.

The average of the AUC of each method is showed in Table 4. The values in this table confirm the superior performance of the hybrid method proposed here.

4. Conclusions

This paper presented an architecture for a system to detect handwritten connected digits using prototype selection schemes. The first one implements prototype selection using self-generation and has few parameters. The second is a training algorithm for NPC that uses a Gaussian Mixture ansatz to describe the data point distributions and uses a method based on stochastic gradient descent to adjust the prototypes in the training process.

The contribution of this work was to apply this hybrid model of prototype selection to the problem of connected digits classification. Besides, a set of 12 numeric features was introduced. The experiments performed over a real-world handwritten digits dataset achieved very good accuracy rates. And the hybrid model achieved superior performance in terms of classification accuracy when compared to the other evaluated methods.

Acknowledgments

This work was supported in part by the Brazilian National Research Council CNPq (Proc. 475911/2008-3) and by FACEPE - Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (Proc. APQ-0890-1.03/08).

References

[1] R. Alhajj and A. Elnagar. Multiagents to separating handwritten connected digits. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 35(5):593–602, 2005.

[2] L. Bottou. Online learning and stochastic approximations. Online Learning in Neural Networks, D. Saad, Ed., Cambridge: Cambridge University Press, 1998.

[3] A. P. Bradley. The Use of The Area Under The ROC Curve in The Evaluation of Machine Learning Algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[4] H. A. Fayed, S. R. Hashem, and A. F. Atiya. Self-generating prototypes for pattern classification. *Pattern Recognition*, 40(5):1498–1509, 2007.

[5] S.-W. Kim and B. J. Oommen. A Brief Taxonomy and Ranking of Creative Prototype Reduction Schemes. *Pattern Analysis and Applications*, 6(3):232–244, 2003.

[6] S. Seo, M. Bode, and K. Obermeyer. Soft nearest prototype classification. *IEEE Transactions on Neural Networks*, 14(2):390–398, 2003.

[7] X. Wang, V. Govindaraju, and S. Srihari. Holistic Recognition of Handwritten Character Pairs. *Pattern Recognition*, 33(12):1967–1973, 2000.