A Simple and Efficient Method for Global Handwritten Word Recognition Applied to Brazilian Bankchecks

Luan Ling Lee
Faculty of Electrical and Computer Engineering
State University of Campinas
lee@decom.fee.unicamp.br

Stela Isernhagen Coelho
Faculty of Electrical and Computer Engineering
State University of Campinas
stela@decom.fee.unicamp.br

Abstract

A simple and efficient method for the recognition of isolated handwritten words in Brazilian bankchecks is proposed. For this end, the global word recognition technique was adopted. It relies mainly on topological features to identify the shape of the digitized word images. A Genetic Algorithm (GA) was used to select those features that best represent each word class and the Nearest Neighbor Method was used as a classification means. The experimental results obtained in our laboratory present a higher average classification rate than a previous work that used the same database.

1. Introduction

Automated unrestricted handwriting recognition has been the target of many studies in the last two decades and remains to be a very challenging and complex task. Working with dynamic or very large vocabularies, the handwritten words can only be recognized if each one of their letters is individually identified. In applications that involve a reduced lexicon size the segmentation paradigm can be substituted by the use of a global word recognition scheme. In this case, both, training and recognition phases don't require the knowledge of segmentation statistics.

The application chosen for validating the recognition method proposed in this work was the recognition of handwritten legal amounts in Brazilian bankchecks. The lexicon size of worded amounts used in Brazilian bankchecks is essentially about forty words. Notice that this lexicon size is larger than those of other languages, such as English (with 32 words) and French (with 25 words).

The objective of this work is to provide a simple but efficient method that uses only the most discriminating features extracted from each word class during bankcheck legal amount recognition. For this purpose, we rely on the Genetic Algorithm for the feature selection.

The tests were performed using two different databases previously used in other works [1, 2, 3, 5]. Thus, we are able to compare the performance of the currently proposed method with that achieved by the previous works suggested in the literature.

2. Proposed System

The handwritten word recognition method is divided into two distinct phases: the training and the testing phase. Figure 1 presents a block diagram scheme of the proposed method illustrating the operation of the training and the testing phase and information exchange and interaction between them.

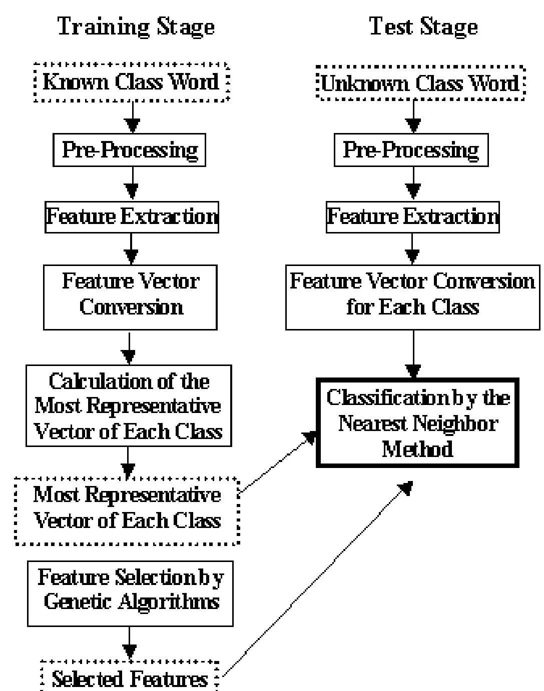

Figure 1. The scheme of the proposed recognition method

Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR'05)
1520-5263/05 $20.00 © 2005 **IEEE**

The training phase begins with the introduction of handwritten words with known and defined word classes. Each input word image is adequately pre-processed and its relevant features are then extracted from the pre-processed word image forming a feature vector. The built feature vector is further converted into another but a small one. This new vector has a variable size according to the currently considered word class. For each word class, a central feature vector is generated, also known as the class reference feature vector, which has the goal of representing its corresponding word class. This class reference feature vector is generated by the application of a genetic algorithm, which has the goal of selecting the most representative features for each word class.

On the other hand, in the testing phase, an image from an unknown class word is initially pre-processed and then having its feature vector extracted. From this initially extracted feature vector, for each distinct word class, a new and different feature vector with reduced size is then generated. As a result, the system has as many new converted feature vectors as the number of word classes, which is the size of the lexicon. Next, the Nearest Neighbor Method is used to select the most appropriated class by calculating the distance between each generated class vector and the corresponding class reference feature vector selected previously by the genetic algorithm during the training stage. In other words, the Nearest Neighbor Method will assign the input handwritten word to the class that best accommodates the input image in terms of the smallest distance between the converted vector and the most representative class vector.

This section is dedicated to a detailed description of each one of these processes involved in the training and the testing stage.

2.1 Word Image Pre-Processing

The image of a handwritten word is subjected to deformations introduced by the process of writing and digitalization. In general, the deformations caused by the writing process are related to the material and instruments used in writing as well as to each subject's writing style. The application of some pre-processing techniques aims to eliminate or reduce these undesirable deformations, facilitating, as a consequence, the task of feature extraction from handwritten word image and minimizing the variations among samples of the same word class, i.e., the intra-variability.

The pre-processing techniques considered and implemented in this work consist of binarization, noise removal, base line skew correction, slant correction and trace thinning.

2.2 Feature Extraction

In our feature extraction process, we focus on a set of global parameters. Some of these parameters are selected from the related literature while some others are of our own contribution, derived from laboratory experience.

Precisely seventeen global features are considered. Table 1 lists all of them.

Table 1. The set of features extracted from input word images.

	Features
1	*ascenders*
2	*descenders*
3	*loops*
4	vertical strokes
5	horizontal strokes
6	south east strokes
7	south west strokes
8	number of *ascenders*
9	number of *descenders*
10	number of *loops*
11	number of vertical strokes
12	number of horizontal strokes
13	number of south east strokes
14	number of south west strokes
15	size of the word
16	quantity of black pixels
17	Word length/word height

Among those features extracted from an input word image, the first seven listed features of Table 1 are called 'composed' features, each one occupying 100 entrances of the entire feature vector. The other ten features are called 'simple' features, each one occupying only one position of the feature vector. As a consequence, the initial feature vector extracted from a word image has 710 elements in total.

To see the generation of a 'composed' feature, we observe the following example. When an ascender is detected from an input word image, its normalized position measured from the left boundary of the word is calculated. In other words, the word length (width) is normalized to 100 pixels in total. When a topological feature, e.g., an ascender, is detected at the 85th pixel, we assign an '1' to that entrance of the extracted word feature vector, which represents a spatial range that covers the 85th pixel.

2.3 Feature Vector Conversion

As mentioned before, the conversion of an extracted feature vector from a word image in the training phase is straightforward because the word class is known a priori.

On the other hand, in the testing phase, since the word class of an input word image is unknown, the extracted feature vector from the input word image needs to be converted (transformed) individually to a reduced-size feature vector for each corresponding word class of the word lexicon. Therefore, the number of generated feature vectors with the reduced size is equal to the number of word classes of the considered word lexicon. Notice that the new reduced feature vectors generated for different word classes have distinct dimensions that increase with the number of alphabet letters in each word.

Let us go back to the previous example where an *ascender* is detected from an input word image at the 85th pixel of the normalized word length. If the extracted feature vector will be compared with a reference feature vector of a determined reference word class, before this the initial extracted feature vector should undergo an adequate conversion. Figure 2 explicitly shows the initial extracted feature vector and the converted feature vectors for word class "um" and "noventa", which consist of two and seven letters, respectively. In other words, 2 and 7 entrances are needed to convert an *ascender* feature for the "um" and "noventa" word classes, respectively. When the *ascender* is detected at the 85th pixel, the relative position of this detected ascender in the converted vector that best accommodates it is the 2nd and the 6th entrance of the corresponding ascender vector segment.

Similarly the same procedure of feature vector conversion is applied to the other six "composed feature" types.

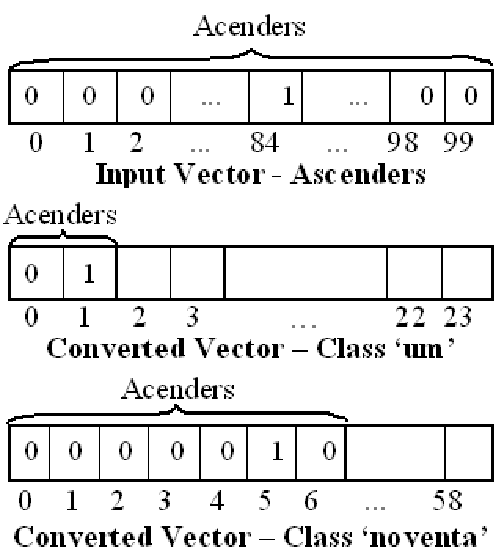

Figure 2. Example of feature vector conversion for an ascender detected at the 85th pixel of the normalized word length.

The size of a converted feature vector is therefore a function of the size of the class word or the number of alphabet letters in the word. Equation 1 is used to calculate the size of a converted feature vector.

$$TamVet\,(i) = [Comp * n(i)] + Simp, \qquad (1)$$

Notice that *TamVet(i)* is the size of the vector for the word class *i*, *Comp* represents the number of 'composed' features, *n(i)* indicates the number of alphabet letters of the word class i, *and* *Simp* is the total number of simple features in the considered word class. For the currently considered example, *Comp* and *Simp* are found constant, equal to seven and ten, respectively.

2.4 Word class representative vectors: reference feature Vectors

During the training phase, a reference feature vector is generated and used to represent the corresponding word class. This reference feature vector is later used in the testing phase for the classification of new, unknown input words. In this work, a reference feature vector is obtained from averaging all converted feature vectors for each class word used in the training stage.

2.5 Feature selection based on the genetic algorithm

In this work a genetic algorithm is used for feature selection in order to retain only those feature elements that help to increase the correct classification rate and to simultaneously eliminate those noisy/redundant feature elements that do not contribute to a better performance. In other words, the Genetic Algorithm (GA) [6, 7] is used as a discriminant feature selector.

The basic idea of the simple GA implemented in this work is to discover the most representative feature elements of each class, in terms of large intra-class similarity as well as large inter-class dissimilarity. As it can be seen, the features cannot be ordered by importance as each word class has its own most important features.

More precisely, the GA was implemented based on the following system configuration: a population composed of 20 individuals, 0.3 crossover probability and 0.1 mutation probability. The results presented here were achieved with the GA lifetime set equal to 1000 generations.

For each class, a vector of '0's and '1's of seventeen (number of features) elements was generated. The '0's indicate the absence of the features represented by that chromosome position and the '1's indicate the presence of these features.

2.6 Classification

The classification method used was based on the statistic Nearest Neighbor. The implemented classification rule consists in comparing the Euclidian distances between an input word feature vector and the most representative vector of each word class. The chosen class will be the one that achieves the smallest Euclidean Distance (D) between the converted input feature vector (VE) and the most representative vector of the corresponding word class (VC), i.e.,

$$D = \sum_{i=0}^{n} \sqrt{(VE_{(i)}^{2}) - (VC_{(i)}^{2})}, \qquad (2)$$

where n is the number of elements of the class vector and i is the position of the current feature element. Notice that only those features selected by the Genetic Algorithm are taken into account for the distance computation and class decision.

3. Experimental investigation

3.1 Databases

Two sets of bankcheck databases were used in this investigation. One database was produced by the Pattern Recognition and Communication Networks Laboratory - LRPRC – UNICAMP – Brazil and had already been extensively used in some previous research works [3, 5]. This database consists of 3178 word images, about 1200 of which were collected from a set of 200 Brazilian bankchecks contributed by a population of 90 people. The rest of word images were collected and digitized in the laboratory from white paper sheets.

The other database, widely used in [1] and [2], was produced by the Document Image Analysis and Recognition Laboratory - LARDOC - PUC-PR – Brazil. This database is composed of 15636 words and 15138 numeral digit collected from 2016 Brazilian bankchecks.

3.2 Investigation results

For the purpose of performance comparison, the word samples of the LRPRC-UNICAMP database were organized exactly as done in [3]. The samples were separated into ten groups and organized for ten experiments. The organization of ten sample groups is as follows. On each run the combination of samples of nine groups were used for training (samples for the calculation of the most representative vector of each class) and the resting one group was used for testing.

Table 2 shows the performance of the proposed recognition method in comparison with that based on HMMs and Fuzzy HMMs obtained in [3] for ten experiments and the overall average performances. Notice that the proposed handwritten word recognition method outperforms that proposed in [3] considerably in all three cases where the correct word class is among N most probable word classes for N=1, 5 and 10.

Table 2. Comparison of the results obtained in [3] and that obtained from applying the proposed method.

Word Recognition Results per Group (% Correct in top N choices)						
	Gomes (2000) Method			Proposed Method		
Groups	**N=1**	**N=5**	**N=10**	**N=1**	**N=5**	**N=10**
1	55	84	95	71.23	92.94	95.33
2	50	83	94	68.96	96.85	99.07
3	50	84	94	61.59	97.65	99.54
4	56	87	95	54.21	97.60	99.69
5	50	84	92	55.09	95.32	99.46
6	53	80	91	60.49	96.62	98.05
7	51	84	97	62.62	96.09	99.19
8	43	80	94	59.56	97.15	99.79
9	46	83	92	60.77	96.24	99.72
10	52	83	93	59.59	94.34	99.58
General Mean	**51**	**83**	**94**	**61.41**	**96.08**	**98.95**

Table 3. Comparison of the Results Obtained in [2] and by the Proposed Method with the LARDOC - PUC-PR database

Words	**PPCCR (%)**	**Proposed Method (%)**	**Words**	**PPCCR (%)**	**Proposed Method (%)**
Um	92.42	95.96	Trinta	47.30	94.60
Dois	72.86	67.62	Quarenta	65.67	75.49
Três	59.70	62.63	Cinqüenta	78.67	85.08
Quatro	81.25	89.80	Sessenta	36.62	55.56
Cinco	77.27	85.30	Setenta	37.84	42.34
Seis	51.67	44.44	Oitenta	55.56	78.38
Sete	43.40	32.95	Noventa	65.08	89.58
Oito	73.77	91.11	Cem/Cento	58.18	69.13
Nove	76.19	86.46	Duzentos	69.39	78.08
Dez	66.67	93.33	Trezentos	53.19	81.16
Onze	50.00	46.67	Quatrocentos	67.44	100.00
Doze	20.00	55.56	Quinhentos	79.17	80.00
Treze	71.43	88.89	Seiscentos	44.19	70.83
Quatorze	28.57	41.67	Setecentos	57.14	58.67
Quinze	55.56	66.67	Oitocentos	66.00	86.67
Dezesseis	16.67	55.56	Novecentos	80.85	78.57
Dezessete	28.57	91.67	Mil	84.82	93.86
Dezoito	66.67	22.22	Reais/Real	67.46	79.00
Dezenove	88.89	83.33	Centavo/ Centavos	83.03	90.58
Vinte	63.16	50.00			
General Mean PPCCR (%)		**67.66**	**General Mean Proposed Method (%)**		**72.72**

In order to compare the proposed recognition method with that implemented by the LARDOC - PUC-PR (PPCCR), the LARDOC - PUC-PR database was used for this part of experimental analysis. The word recognition method (PPCCR), proposed by the LARDOC - PUC-PR, organizes the word samples of the LARDOC - PUC-PR database in the following manner: 60% for training, 20% for validation and 20% for test [2]. In other words, the PPCCR method has an additional system validation phase, besides the training and the testing stage. Since our work does not have the validation phase, the samples were divided only into two groups: 70% for training (samples for the calculation of the most representative vector of each class) and 30% for test.

Table 3, comparatively shows the performance of the PPCCR reported in [2] and that achieved by the proposed method for each word class as well as the overall average recognition rate. In summary, the proposed method in this work outperforms the PPCCR proposed in [2] considerably.

4. Conclusion

Regarding the results presented in Table 3, we can notice that both methods, the proposed one and that in [2], have difficulty to achieve a good recognition rate for some word classes. More precisely, the word class that present the lowest classification rates for the PPCCR method proposed in [2] is the 'dezesseis' word class while in the proposed method the worst case is the 'dezoito' word class. A little thinking can help us to realize that word classes such as 'dezesseis', 'dezessete' and 'dezoito' have many common alphabet letters, and consequently share many similar features. Notice that these common features contribute considerably to the level of confusion among them; therefore, classify these word classes correctly remains a challenging task.

In spite of these difficulties, we can notice that some significant improvements were achieved by the proposed method. Another very important thing is that the proposed recognition method was validated through two distinct database in comparison with two previous approaches closely related to the theme of Brazilian bankcheck recognition.

With respect to the approaches suggested in the literature of bankcheck recognition for other languages such as English and French, that apparently are capable of providing better performance, suitable and careful comparison schemes are needed. This sort of comparison is by no mean trivial due to the fact of a remarkable difference in the lexicon size as well as its contents. In fact, the lexicon sizes for legal amount in English and French are considerably smaller than that in Portuguese, facilitating the correct recognition rate.

Probably a major merit of the proposed word recognition method is its simplicity without losing its high classification capability.

5. References

[1] C.O.A., M. Freitas, Morita, L.E.S. de Oliveira, E. Justino, A. El-Yacoubi, E. Lethelier, F. Bortolozzi, and R. Sabourin, Databases for Brazilian Bankcheck Recognition In: XXVI Conferência Latino-americana de Informática, 2000. (In Portuguese)

[2] Freitas, C.O.A. The Use of Hidden Markov Models for Handwritten Word Recognition. *PhD Thesis*, Pontifícia Universidade Católica do Paraná, Curitiba, Brasil, 2002. (In Portuguese)

[3] N.R. Gomes, Handwritten Word Recognition Based on HMM and Topological and Geometric Feature. *PhD Thesis*, Faculdade de Engenharia Elétrica da Universidade Estadual de Campinas, Brasil, 2000. (In Portuguese)

[4] D. Guillevic, Unconstrained Handwritten Recognition Applied to the Processing of Bank Checks. *PhD Thesis,* Concordia University, Montreal, Quebec, Canadá, 1995.

[5] A. Koerich, and L.L. Lee. Automatic Bankcheck Processing: Information Storage and Recovery. Revista Controle & Automação, Vol 12 nº 1, 2001. (In Portuguese)

[6] Z. Michalewick, Genetic Algorithms + Data Structures = Evolution Programs. 3rd edition, Springer, 1996.

[7] M. Mitchel, *An Introducing To Genetic Algorithms.* MIT Press, 1996.