Recognition and Verification of Unconstrained Handwritten Words

Alessandro L. Koerich, *Member*, *IEEE*, Robert Sabourin, *Member*, *IEEE*, and Ching Y. Suen, *Fellow*, *IEEE*

Abstract—This paper presents a novel approach for the verification of the word hypotheses generated by a large vocabulary, offline handwritten word recognition system. Given a word image, the recognition system produces a ranked list of the *N*-best recognition hypotheses consisting of text transcripts, segmentation boundaries of the word hypotheses into characters, and recognition scores. The verification consists of an estimation of the probability of each segment representing a known class of character. Then, character probabilities are combined to produce word confidence scores which are further integrated with the recognition scores produced by the recognition system. The *N*-best recognition hypothesis list is reranked based on such composite scores. In the end, rejection rules are invoked to either accept the best recognition hypothesis of such a list or to reject the input word image. The use of the verification approach has improved the word recognition rate as well as the reliability of the recognition system, while not causing significant delays in the recognition process. Our approach is described in detail and the experimental results on a large database of unconstrained handwritten words extracted from postal envelopes are presented.

Index Terms—Word hypothesis rejection, classifier combination, large vocabulary, handwriting recognition, neural networks.

1 Introduction

Recognition of handwritten words has been a subject of intensive research in the last 10 years [1], [2], [3], [4], [5], [6], [7], [8]. Significant improvements in the performance of recognition systems have been achieved. Current systems are capable of transcribing handwriting with average recognition rates of 50-99 percent, depending on the constraints imposed (e.g., size of vocabulary, writer-dependence, writing style, etc.) and also on the experimental conditions. The improvements in performance have been achieved by different means. Some researchers have combined different feature sets or used optimized feature sets [7], [9], [10]. Better modeling of reference patterns and adaptation have also contributed to improve the performance [2], [7]. However, one of the most successful approaches to achieving better performance is the combination of classifiers. This stream has been used especially in application domains where the size of the lexicon is small [1], [11], [12], [13], [14], [15]. Combination of classifiers relies on the assumption that different classification approaches have different strengths and weaknesses which can compensate for each other through the combination.

- *A.L. Koerich is with the Postgraduate Programme in Applied Informatics (PPGIA), Pontifical Catholic University of Paraná (PUCPR), R. Imaculada Conceição, 1155, Curitiba, PR, 80215-901, Brazil. E-mail: alekoe@computer.org.*
- *R. Sabourin is with the Departement de Génie de la Production Automatisée (GPA), École de Technologie Supérieure, 1100 rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3, Canada. E-mail: robert.sabourin@etsmtl.ca.*
- *C.Y. Suen is with the Centre for Pattern Recognition and Machine Intelligence (CENPARMI), Concordia University, 1445 de Maisonneuve Blvd. West, VE 3180, Montréal, Québec, H3G 1M8, Canada. E-mail: suen@cenparmi.concordia.ca.*

Manuscript received 14 Oct. 2003; revised 17 Dec. 2004; accepted 17 Jan. 2005; published online 11 Aug. 2005.
Recommended for acceptance by K. Yamamoto.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0317-1003.

Verification can be considered as a particular case of combination of classifiers. The term verification is encountered in other contexts, but there is no consensus about its meaning. Oliveira et al. [16] define verification as the postprocessing of the results produced by recognizers. Madhvanath et al. [17] define verification as the task of deciding whether a pattern belongs to a given class. Cordella et al. [18] define verification as a specialized type of classification devoted to ascertaining in a dependable manner whether an input sample belongs to a given category. Cho et al. [19] define verification as the validation of hypotheses generated by recognizers during the recognition process. In spite of different definitions, some common points can be identified and a broader definition of verification could be a postprocessing procedure that takes as input hypotheses produced by a classifier or recognizer and which provides as output a single reliable hypothesis or a rejection of the input pattern. In this paper, the term verification is used to refer to the postprocessing of the output of a handwriting recognition system resulting in rescored word hypotheses.

In handwriting recognition, Takahashi and Griffin [20] are among the earliest to mention the concept of verification and the goal was to enhance the recognition rate of an OCR algorithm. They have designed a character recognition system based on a multilayer perceptron (MLP) which achieves a recognition rate of 94.6 percent for uppercase characters of the NIST database. Based on an error analysis, verification by linear tournament with one-to-one verifiers between two categories was proposed and such a verification scheme increased the recognition rate by 1.2 percent. Britto et al. [21] used a verification stage to enhance the recognition of a handwritten numeral string HMM-based system. The verification stage, composed of 20 numeral HMMs, has improved the recognition rate for strings of different lengths by about 10 percent (from 81.65 percent to 91.57 percent). Powalka et al. [11] proposed a hybrid recognition system for online handwritten word recognition where letter verification is introduced to improve disambiguation among word hypotheses.

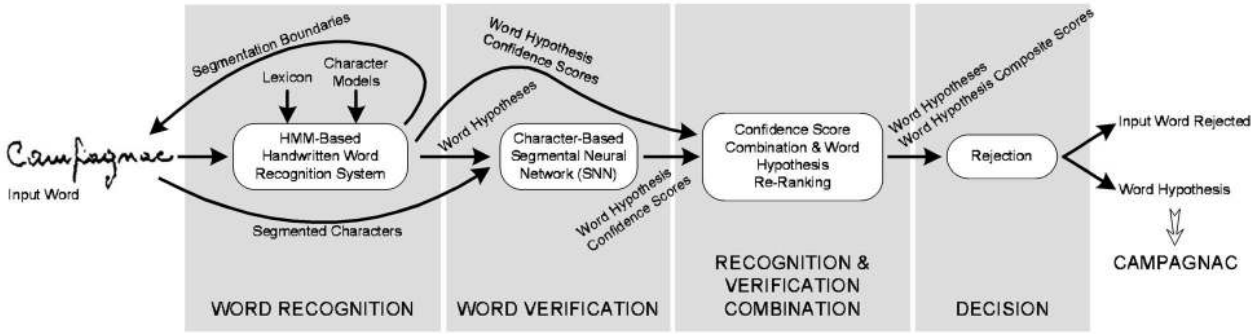


Fig. 1. An overview of the main components of the recognition and verification approach: HMM-based word recognition, character-based word verification, combination of recognition and verification confidence scores, and word hypothesis rejection.

A multiple interactive segmentation process identifies parts of the input data which can potentially be letters. Each potential letter is recognized and further concatenated to form strings. The letter verification procedure produces a list of words constrained to be the same words provided by a holistic word recognizer. Scores produced by the word recognizer and by the letter verifier are integrated into a single score using a weighted arithmetic average. Improvements between 5 percent and 12 percent in the recognition rate are reported. Madhvanath et al. [17] describe a system for rapid verification of unconstrained offline handwritten phrases using perceptual holistic features. Given a binary image and a verification lexicon containing ASCII strings, holistic features are predicted from the verification ASCII strings and matched with the feature candidates extracted from the binary image. The system rejects errors with 98 percent accuracy at a 30 percent acceptance level. Guillevic and Suen [22] presented a verification scheme at character level for handwritten words from a restricted lexicon of legal amounts of bank checks. Characters are verified using two k-NN classifiers. The results of the character recognition are integrated with a word recognition module to shift up and down word hypotheses to enhance the word recognition rate.

Some works give a different meaning to verification and attempt to improve reliability. Recognition rate is a valid measure to characterize the performance of a recognition system, but, in real-life applications, systems are required to have a high reliability [23], [24], [25], [26]. Reliability is related to the capability of a recognition system not to accept false word hypotheses and not to reject true word hypotheses. Therefore, the question is not only to find a word hypothesis, but also to find out the trustworthiness of the hypothesis provided by a handwriting recognition system. This problem may be regarded being as difficult as the recognition itself. It is often desirable to accept word hypotheses that have been decoded with sufficient confidence. This implies the existence of a hypothesis verification procedure which is usually applied after the classification.

Verification strategies whose only goal is to improve reliability usually employ mechanisms that reject word hypotheses according to established thresholds [23], [24], [25], [26]. Garbage models and antimodels have also been used to establish rejection criteria [4], [25]. Pitrelli and Perrone [26] compare several confidence scores for the verification of the output of an HMM-based online handwriting recognizer. Better rejection performance is achieved by an MLP classifier that combines seven different confidence measures. Marukatat et al. [25] have shown an efficient measure of confidence for an online handwriting recognizer based on antimodel measures which improves the recognition rate from 80 percent to 95 percent at a 30 percent rejection level. Gorski [24] presents several confidence measures and a neural network to either accept or reject word hypothesis lists. Such a rejection mechanism is applied to the recognition of courtesy check amount to find suitable error/rejection trade-offs. Gloger et al. [23] presented two different rejection mechanisms, one based on the relative frequencies of reject feature values and another based on a statistical model of normal distributions to find a best trade-off between rejection and error rate for a handwritten word recognition system.

This paper is focused on a method of improving the performance of an existing state-of-the-art offline handwritten word recognition system which is writer-independent for very large vocabularies. The challenge is to improve the recognition rate while not increasing the recognition time substantially. To achieve such a goal, a novel verification approach which focuses on the strengths and weaknesses of an HMM classifier is introduced. The goal of the verification approach is to emphasize the strengths of the HMM approach to alleviate the effects of its shortcomings in order to improve the overall recognition process. Furthermore, the verification approach is required to be fast enough without delaying the recognition process. The handwriting recognition system is based on a hidden Markov model and it provides a list of the N-best word hypotheses, their a posteriori probabilities, and characters segmented from such word hypotheses. The verification is carried out at the character level and a segmental neural network is used to assign probabilities to each segment that represents a character. Then, character probabilities are averaged to produce word confidence scores which are further combined with the scores produced by the handwriting recognition system. The N-best word hypotheses are reranked based on such composite scores and, at the end, rejection rules are invoked to either accept or reject the best word hypothesis of such a list. An overview of the main components of the recognition and verification approach is shown in Fig. 1.

The novelty of this approach relies on the way verification is carried out. The verification approach uses the segmentation hypotheses provided by the HMM-based recognition system, which is one of the strengths of the HMM approach, and goes back to the input word image to extract new features from the segments. These feature vectors represent whole

characters and are rescored by a neural network. The novelty also relies on a combination of the results from two different representation spaces, i.e., word and character to improve the overall performance of the recognition process. Some other contributions of this paper include the rejection process and an analysis of the speed, which is crucial when dealing with large vocabularies.

The rest of this paper is organized as follows: Section 2 presents an overview of the HMM-based handwritten word recognition system to provide some minimal understanding of the context in which word verification is applied. Section 3 presents the basic idea underlying the verification approach and its combination with the handwriting recognition system. Rejection strategies are presented in Section 4. Section 5 reports experiments and results obtained and some conclusions are drawn in the last section.

2 Handwriting Recognition System (HRS)

Our system is a large vocabulary offline handwritten word recognition based on discrete hidden Markov models. The HRS was designed to deal with unconstrained handwriting (handprinted, cursive, and mixed styles), multiple writers (writer-independent), and dynamically generated lexicons. Each character is modeled by a 10-state left-to-right transition-based discrete HMM with no self-transitions. Intraword and interword spaces are modeled by a two-state left-to-right transition-based discrete HMM [4].

The HRS includes preprocessing, segmentation, and feature extraction steps (top of Fig. 3). The preprocessing stage eliminates some variability related to the writing process and that is not very significant from the viewpoint of recognition, such as the variability due to the writing environment, writing style, acquisition, and digitization of image, etc. The segmentation method performs an explicit segmentation of the words that deliberately proposes a high number of segmentation points, offering, in this way, several segmentation options, the best ones to be validated during recognition. This strategy may produce correctly segmented, undersegmented, or oversegmented characters. Unlike isolated character recognition, lexicon-driven word recognition approaches do not require features to be very discriminating at the character level because other information, such as context, word length, etc., are available and permit high discrimination of words. Thus, features at the grapheme level are considered with the aim of clustering letters into classes. A grapheme may consist of a full character, a fragment of a character, or more than a character. The sequence of segments obtained by the segmentation process is transformed into a sequence of symbols by considering two sets of features where the first set is based on global features and the second set is based on an analysis of the two-dimensional contour transition histogram of each segment in the horizontal and vertical directions. There are also five segmentation features, that try to reflect the way segments are linked together. The output of the feature extraction process is a pair of symbolic descriptions of equal length, each consisting of an alternating sequence of segment shape symbols and associated segmentation point symbols [4].

2.1 Recognition

The general problem of recognizing a handwritten word w or, equivalently, a character sequence constrained to spellings in a lexicon $\mathcal{L}$, is typically framed from a statistical perspective where the goal is to find the sequence of labels $c_1^L = (c_1 c_2 \dots c_L)$ (e.g., characters) that is most likely, given a sequence of T discrete observations $o_1^T = (o_1 o_2 \dots o_T)$:

$$\hat{w} \ni P\left(\hat{w}|o_1^T\right) = \max_{w \in \mathcal{L}} P\left(w|o_1^T\right). \tag{1}$$

The posteriori probability of a word w can be rewritten using Bayes' rule:

$$P\left(w|o_1^T\right) = \frac{P\left(o_1^T|w\right)P(w)}{P\left(o_1^T\right)}, \tag{2}$$

where $P(w)$ is the a priori probability of the word occurring, which depends on the vocabulary used and the frequency counts in the training data set. The probability of data occurring $P(o_1^T)$ is unknown, but assuming that the word w is in the lexicon $\mathcal{L}$ and that the decoder computes the likelihoods of the entire set of possible hypotheses (all lexicon entries), then the probabilities must sum to one:

$$\sum_{w \in \mathcal{L}} P\left(w|o_1^T\right) = 1. \tag{3}$$

In such a way, estimated posterior probability can be used as confidence estimates [27]. We obtain the posterior $P(w|o_1^T)$ for the word hypotheses analogous to [2], [26], [27]:

$$P(w|o_1^T) = \frac{P(o_1^T|w)P(w)}{\sum\limits_{w \in \mathcal{L}} P(o_1^T|w)P(w)}. \tag{4}$$

2.2 Output of the Handwriting Recognition System

The HRS generates a list of the N-best recognition hypotheses ordered according to the a posteriori probability assigned to each word hypothesis. Each recognition hypothesis consists of:

- a text transcript (H_n), which is given as a sequence of characters $H_n = (c_1^n c_2^n \dots c_L^n)$ in which L is the number of characters in the word;
- segmentation boundaries of the word hypothesis into characters (S_n) which are obtained by the backtracking of the best state sequence by the decoding algorithm [28], [29]. It is given as a sequence of L segments $S_n = (x_1^n x_2^n \dots x_L^n)$, where each segment in S_n corresponds to a character in H_n;
- a recognition score in the form of a posteriori probability which is computed according (4) and further normalized to confidence score by (9).

2.3 Motivation

When passing from recognition to verification, a good knowledge of the behavior of the recognition systems is required. It is important to identify the strengths of the approach and the possible sources of errors to propose novel approaches that are able to minimize such errors. In this way, the motivations for using HMMs in handwriting recognition are: 1) HMMs can absorb a lot of variability related to the intrinsic nature of handwriting, 2) HMMs are very good for the localization of the characters within word, and 3) the truth word hypothesis is frequently among the best word hypotheses. These are some of the reasons why HMM approaches are prevalent in handwriting and speech recognition [2], [3], [5], [7], [30].

In spite of the suitability of the HMMs to the problem in question, there are also some shortcomings associated with such an approach. Two distinct studies conducted to identify the sources of errors and confusions in the HRS have shown that many confusions arise from the features extracted from the loose-segmented characters [9], [31]. These features cannot give a good description of characters when oversegmentation occurs, especially for cursive words. The reason is that the feature set used to characterize cursive words is based on topological information such as loops and crossings which are difficult to be detected from the character fragments.

The errors are also related to the shortcomings associated with the Markovian modeling of the handwriting. The first shortcoming is the susceptibility to modeling inaccuracies that are associated with HMM character models. It is often the case that local mismatches between the handwriting and HMM model can have a negative effect on the accumulated score used for making a global decision. The second shortcoming is the limitation of the HMMs to model the handwriting signal: The assumption that neighboring observations are conditionally independent prevents an HMM from taking full advantage of the correlation that exists among the observations of a character [30].

3 Verification

The goal of the verification is to exploit the strengths of the HMMs to overcome the shortcomings in order to improve the recognition rate of the HRS under the constraint of not increasing the recognition time. The approach we have selected for this is to develop the concept of segmental neural networks as proposed in speech recognition by Zavaliagkos et al. [30]. A segmental neural network is a neural network that estimates a posteriori probability for a complete character segment as a single unit, rather than a sequence of conditionally independent observations. Furthermore, the verification approach was chosen to be modular and applied only to the output of the recognition system. Such a modular approach is suitable for further evaluating the effects of the verification on the overall recognition performance.

Besides exploiting the shortcomings of the HMMs, another motivation to use such an approach is the difference in the recognition rates when considering only the best word hypothesis and the first 10-best word hypotheses, which is greater than 15 percent for an 80,000-word lexicon. This difference is attributed to the presence of very similar words in the lexicon that may differ only by one or two characters. The proposed verification approach is expected to better discriminate between characters and to alleviate this problem.

3.1 Methodology

The main assumption in designing the verification approach is that the segmentation of the words into characters carried out by the HRS is reliable. This assumption is based on previous studies that have shown that the segmentation is reliable in most of the cases [9], [31], [32]. For instance, a visual inspection carried out on 10,006 word images from the training data set of the SRTP[1] database together with a comparison with the alignment (segments-characters) provided by the HRS has identified segmentation problems in less than 20 percent of the cases (2,001 out of 10,006 words).

1. Service de Recherche Technique de la Poste, France.

Fig. 2. Character boundaries (S_n) and text transcripts (H_n) for the 10-best recognition hypotheses generated by the HRS to the input image *Campagnac*.

We rely on the segmentation of the word hypotheses into characters (S_n) and on their labels (H_n) to build a segmental neural network to carry out verification at the character level in an attempt to better discriminate between characters and reduce the ambiguity between similar words. Given the output of the HRS, character alternatives are located within the word hypotheses by using the character boundaries, as shown in Fig. 2. Another module is used to extract new features from such segments and the task of the segmental neural network is to assign a posteriori probabilities to the new feature vectors representing isolated characters, given that their classes are known a priori. Further, the probabilities of the individual characters can be combined to generate confidence scores to the word hypotheses in the N-best word hypothesis list. Then, these confidence scores are combined with the recognition scores provided by HRS through a suitable rule to build a recognition and verification system. Fig. 3 presents an overview of the integration of the modules of the handwriting recognition system, the verification stage, the combination stage, and the decision stage. The following sections describe the main components of the verification stage and how they are built and integrated into the handwriting recognition system.

3.2 Estimation of Confidence Scores for the Word Hypotheses

The verification scheme is based on the estimation of character probabilities by a multilayer perceptron (MLP) neural network which is called segmental neural network (SNN). The architecture of the SNN resembles a standard MLP character classifier. The task of the SNN is to assign an a posteriori probability to each segment representing a character given that the character class has already been assigned by the HRS. We define x_l as the feature vector corresponding to the lth word segment and c_l as the character class of the lth word segment provided by the HRS. The output of the SNN is $P(c_l|x_l)$, which is the a posteriori probability of the character class c_l given the feature vector x_l.

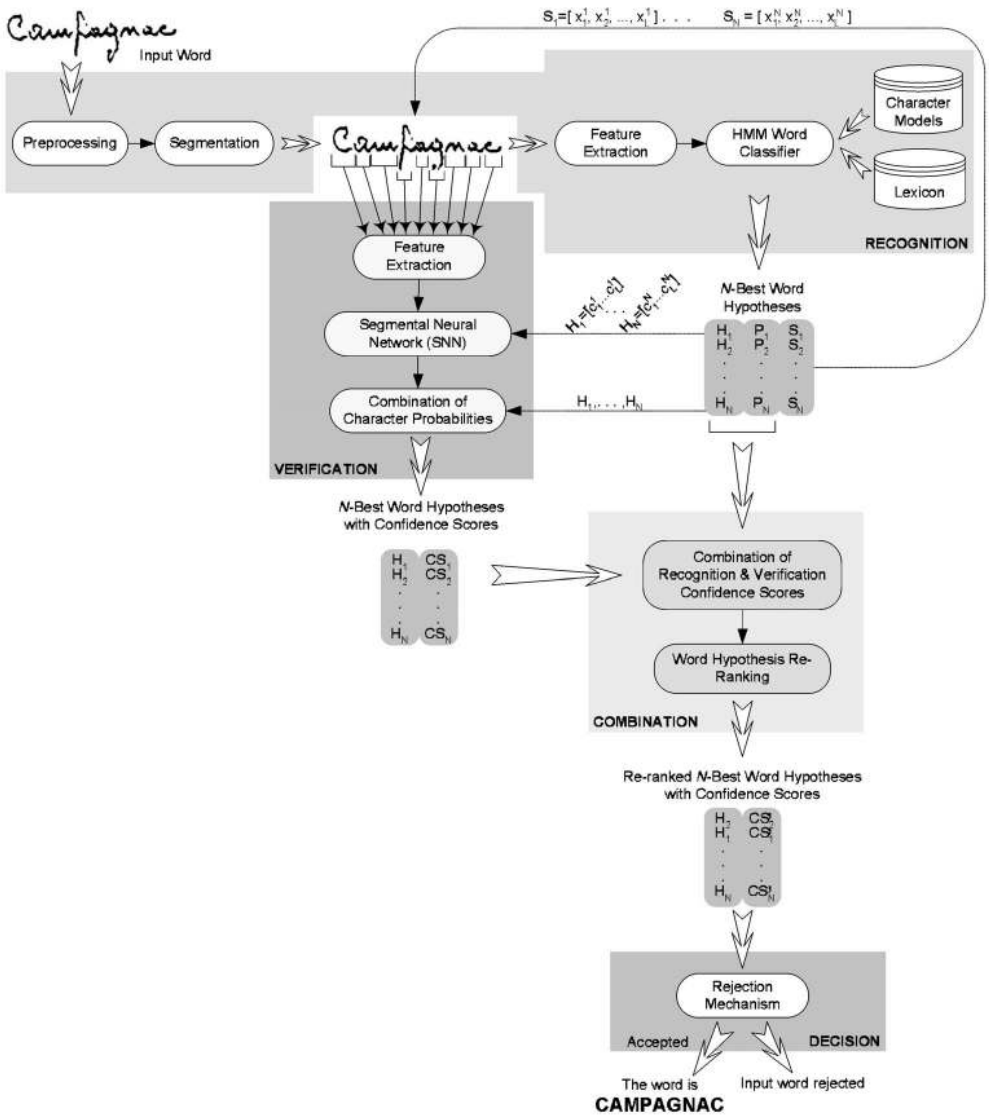


Fig. 3. An overview of the integration of the handwriting recognition system and verification, combination, and decision stages.

To build the SNN, we have considered a 26-class problem, where uppercase and lowercase representations of characters are merged into a unique class called *metaclass* (e.g., "A" and "a" form the metaclass "Aa"). The main reason for such a choice is the weakness of the HRS in distinguishing between uppercase and lowercase characters (only 45 percent of the character cases are recognized correctly). The network takes a 108-dimensional feature vector as input and it has 100 units in the hidden layer and 26 outputs, one for each character class.

The isolated characters are represented in the feature space by 108-dimensional feature vectors which are formed by combining three different types of features: projection histogram from whole characters, profiles from whole characters, and directional histogram from six zones. These features were chosen among others through an empirical evaluation where the recognition rate and the feature vector dimension were used as criteria [33].

3.2.1 Frequency Balancing

The training data exhibits very nonuniform priors for the various character classes and neural networks readily model these priors. However, reducing the effects of these priors on the network, in a controlled way, forces the network to allocate more resources to low-frequency, low-probability classes [34]. This is of significant benefit to the recognition process. To this end, the frequency of the character class during training is explicitly balanced by skipping and repeating samples, based on a precomputed repetition factor, as suggested by Yaeger et al. [34]. Each presentation of a repeated sample is "warped" randomly, which consists of small changes in size, rotation, and horizontal and vertical scalings [33].

3.2.2 Correction of A Priori Character Class Probabilities

Networks with outputs that estimate Bayesian probabilities do not explicitly estimate the three terms on the right of (5) separately.

$$P(c_l|x) = \frac{P(x|c_l)P(c_l)}{P(x)}. \quad (5)$$

However, for a given character c_l, the output of the network, denoted by $P(c_l|x)$, is implicitly the corresponding

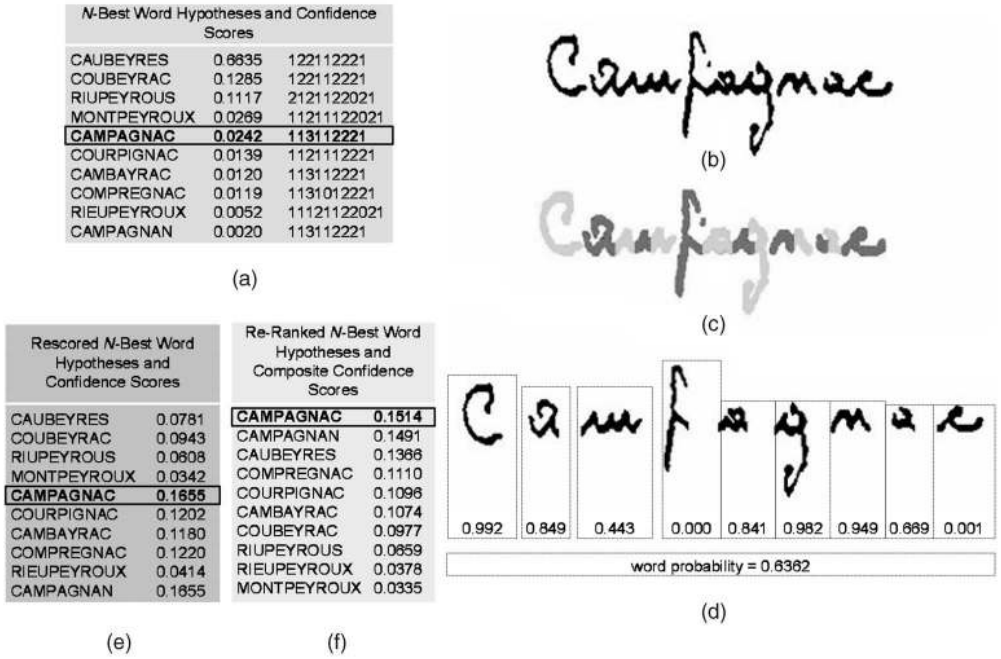


Fig. 4. (a) List of the *N*-best word hypotheses generated by the HRS with the confidence scores and segmentation boundaries. (b) Input handwritten word. (c) Loose segmentation of the word into characters produced by the segmentation step. (d) Final segmentation of the word into characters according to the fifth best recognition hypothesis and the character probabilities estimated by the SNN. (e) Confidence scores estimated by the VS for each word hypothesis. (f) Reranked word hypotheses based on the composite confidence scores obtained by the weighted sum rule.

a priori class probability $P(c_l)$ times the class probability $P(x|c_l)$ divided by the unconditional input probability $P(x)$.

During the network training process, a priori class probabilities $P(c_l)$ were modified due to frequency balancing. As a result, the output of the network has to be adjusted to compensate for training data with class probabilities that are not representative of the real class distributions. Correct class probabilities can be used by first dividing the network outputs by training-data class probabilities and then multiplying by the correct class probabilities:

$$P_{CORR}(c_l|x) = \frac{P(c_l|x)}{P_{TRAIN}(c_l)} P_{REAL}(c_l), \tag{6}$$

where $P_{CORR}(c_l|x)$ denotes the corrected network output, $P_{TRAIN}(c_l)$ denotes the a priori class probability of the frequency-balanced training set, and $P_{REAL}(c_l)$ denotes the real a priori class probability of the training set.

3.2.3 Combination of Character Probabilities

Having a word hypothesis and the probability of each character that forms such a word, it is possible to combine such probabilities to obtain a probability for the word hypothesis. Assuming that the representations of each character are conditionally statistically independent, character estimations are combined by a product rule to obtain word probabilities:

$$\begin{aligned} P(H_n|S_n) &= P(H_n|x_1^n \dots x_l^n) = P(c_1^n \dots c_l^n|x_1^n \dots x_l^n) \\ &= \prod_{l=1}^{L} P_{CORR}(c_l|x_l), \end{aligned} \tag{7}$$

where $P(H_n|S_n)$ is the probability of the word hypothesis H_n given a sequence of segments S_n, $P_{CORR}(c_l|x_l)$ is the a posteriori probability estimated by the neural network to each segment x_l, and L is the number of characters at the word hypothesis H_n. However, in practice, this is a severe rule of fusing the character probabilities as it is sufficient for a single character to exhibit a low probability (close to zero) to flaw the word probability estimation [12]. Since we have equal prior word probabilities, an alternative to the product rule is the median rule [12], which computes the average probability as:

$$P(H_n|S_n) = \frac{1}{L} \sum_{l=1}^{L} P_{CORR}(c_l|x_l). \tag{8}$$

This combination scheme is similar to that proposed in [11] and it is illustrated in Fig. 4d. In [35], we have investigated other combination rules.

3.2.4 Character Omission

The architecture of the HMMs in the HRS includes null transitions that model undersegmentation or the absence of a character within a word (usually due to the misspelling of the word by the writer) [36]. This situation occurs in Fig. 2 where, for some word hypotheses, a segment is not associated with a character label. How should the verification stage behave in such a situation? We have analyzed different ways of overcoming such a problem: ignore the character during the concatenation of characters or assign an average probability to such a character. The former has produced the best results. It consists of computing an average probability score for each character class which is obtained by using the SNN as a standard classifier. For each character class, an "average probability" is computed considering only the inputs that are correctly recognized by the SNN on a validation data set. These average probabilities are used when no segment is associated with a character within the word labels due to undersegmentation problems.

TABLE 1
Rules Used to Combine the Confidence Scores of the Word Hypotheses Provided by the HRS and by the VS

Combination Rules	Definition
Max	$\max[CS_{HRS}(H_n), CS_{VS}(H_n)]$
Sum	$CS_{HRS}(H_n) + CS_{VS}(H_n)$
Product	$CS_{HRS}(H_n) * CS_{VS}(H_n)$
Weighted Sum	$\alpha CS_{HRS}(H_n) + \beta CS_{VS}(H_n)$
Weighted Product	$CS_{HRS}(H_n)^{\alpha} * CS_{VS}(H_n)^{\beta}$

TABLE 2
Different Rejection Rules Used with the Reranked *N*-Best Word Hypothesis Lists at the Decision Stage

Rejection Threshold	Definition
R_{FIXED}	F
R_{AVG_CLASS}	$\frac{1}{K}\sum_{k=1}^{K}\frac{P(w_k\|o_1^t(k))}{\sum_{n=1}^{N}P(w_n\|o_1^t(k))}$
R_{AVG_TOP}	$\frac{1}{N}\sum_{n=1}^{N}CS'(H_n)$
R_{DIF_12}	$CS'(H_1) - CS'(H_2)$

3.3 Combination of Recognition and Verification Scores

We are particularly interested in combining the outputs of the HRS and the VS with the aim of compensating for the weaknesses of each individual approach to improve the word recognition rate. The HRS is regarded as a sophisticated classifier that executes a huge task of assigning a class (a label from $\mathcal{L}$) to the input pattern. On the other hand, the SNN uses quite different features and classification strategy and it works at a different decision space. Improvements in the word recognition rate are expected by combining such different approaches because the verification is focusing on the shortcomings of the HMM modeling. However, before the combination, the output of both HRS and VS should be normalized by (9) so that the scores assigned to the *N*-best word hypotheses sum up to one.

$$CS_{NORM}(H_n) = \frac{P(.|.)}{\sum_{n=1}^{N} P(.|.)}, \tag{9}$$

where $CS_{NORM}(H_n)$ denotes the normalized confidence score for the word hypothesis, $P(.|.)$ corresponds to either $P(w_n|o_1^T)$ or $P(H_n|x_1^L)$, depending on whether the output of the HRS or the output of the VS is being normalized, respectively.

Different criteria can be used to combine the outputs of a HRS and a VS [13]. Gader et al. [1], [15] compare different decision combination strategies for handwritten word recognition using three classifiers that produce different confidence scores. In our case, the measures produced by the classifiers are confidence scores for each word hypothesis in the *N*-best word hypothesis list. From the combination point of view, we have two classifiers, each producing N consistent measurements, one for each word hypothesis in the *N*-best word hypothesis list. The simplest means of combining them to obtain a composite confidence score is by *Max*, *Sum*, and *Product* rules, which are defined in Table 1. In Table 1, $CS_{HRS}(H_n)$ and $CS_{VS}(H_n)$ denote the confidence scores of the HRS and the VS to the n best word hypothesis, respectively. The composite score resulting from the combination is denoted by $CS'(H_n)$.

The basic combination operators do not require training and do not consider differences in the performance of the individual classifiers. However, in our case, the classifiers are not conditionally statistically independent since the space of the VS is a subspace of the HRS. It is logical to introduce weights to the output of the classifiers to indicate the performance of each classifier. Changing the weights allows us to adjust the influences of the individual recognition scores on the final score. Table 1 shows two weighted rules that are also used to combine the confidence scores, where α and β are the weights associated with the HRS and the VS, respectively.

After the combination of the confidence scores of each word hypothesis, the *N*-best word hypothesis list can be ordered again based on such composite scores. Fig. 4f shows an example of the combination of confidence scores using the weighted sum rule for several word hypotheses.

4 Decision

After the verification and combination of confidence scores there is still a list with *N*-best word hypotheses. In real-life applications, the recognition system has to come up with a single word hypothesis at the output or a rejection of the input word if it is not certain enough about the hypothesis.

The concept of rejection admits the potential refusal of a word hypothesis if the classifier is not certain enough about the hypothesis. In our case, the confidence scores assigned to the word hypotheses give evidence about the certainty. Assuming that all words are present in the lexicon, the refusal of a word hypothesis may be due to two different reasons:

- There is not enough evidence to come to a unique decision since more than one word hypothesis among the *N*-best word hypotheses appear adequate.
- There is not enough evidence to come to a decision since no word hypothesis among the *N*-best word hypotheses appears adequate.

In the first case, it may happen that the confidence scores do not indicate a unique decision in the sense that there is not just one confidence score exhibiting a value close to one. In the second case, it may happen that there is no confidence score exhibiting a value close to one. Therefore, the confidence scores assigned to the word hypotheses in the *N*-best word hypothesis list should be used as a guide to establish a rejection criterion.

The Bayes decision rule already embodies a rejection rule, namely, find the maximum of $P(w|o)$ but check whether the maximum found exceeds a certain threshold value or not. Due to decision-theoretic concepts, this reject rule is optimum for the case of insufficient evidence if the closed-world assumption holds and if the a posteriori probabilities are known [37]. This suggests the rejection of a word hypothesis if the confidence score for that hypothesis is below a threshold. In the context of our recognition and verification system, the task of the rejection mechanism is to decide whether the best word hypothesis (*TOP* 1) can be accepted or not (Fig. 3). For such an aim, we have investigated different rejection strategies: class-dependent rejection, where the rejection threshold depends on the class of the word (R_{AVG_CLASS}); hypothesis-dependent rejection, where the rejection threshold depends on the confidence scores of the word hypotheses at the *N*-best list (R_{AVG_TOP}, R_{DIF_12}); global threshold that depends neither on the class nor on the hypotheses (R_{FIXED}). These different thresholds are computed as shown in Table 2,

where K is the number of times the word w_k appears in the training data set and $F \in [0,1]$ is a fixed value. More details about these rejection strategies can be found in [38].

The rejection rule is given by:

1. The *TOP* 1 word hypothesis is accepted whenever

$$CS'(H_1) \geq \gamma R_{(.)}. \quad (10)$$

2. The *TOP* 1 word hypothesis is rejected whenever

$$CS'(H_1) < \gamma R_{(.)}, \quad (11)$$

where H_1 is the best word hypothesis, $R_{(.)}$ is one of the rejection thresholds defined in Table 2, CS' is the composite confidence score obtained by combining the outputs of the HRS and the VS, and $\gamma \in [0,1]$ is a threshold that indicates the amount of variation of the rejection threshold and the best word hypothesis. The value of γ is set according to the rejection level required.

5 Experiments and Results

During the development phase of our research, we have used the SRTP database, which is a proprietary database composed of more than 40,000 binary images of real postal envelopes digitized at 200 dpi. From these images, three data sets that contain city names manually located on the envelopes were generated: a training data set with 12,023 words, a validation data set with 3,475 words, and a test data set with 4,674 words.

In developing the verification stage, the first problem that we have to deal with is to build a database of isolated characters since, in the SRTP database, only the fields of the envelopes are segmented. Furthermore, only the words are labeled and the information related to the segmentation of the words into characters is not available. To obtain the segmentation boundaries and build a character database, a bootstrapping of the HRS was used. The HRS was used to segment the words from the training data set into characters and to label each segment in an automatic fashion. To increase the quality of the samples in the character database, only the characters segmented from words correctly recognized were considered. The procedure to build the database is described in [33].

We have carried out four different types of experiments: recognition of handwritten words, recognition of isolated handwritten characters, combination of recognition and verification results to optimize the overall recognition rate, and rejection of word hypotheses to improve the overall reliability of the recognition process. However, in this paper, we focus on the experiments related to the recognition, verification, and rejection of handwritten words. The experiments related to the recognition of isolated handwritten characters are described in [33]. To evaluate the results, the following measures are employed: recognition rate, error rate, rejection rate, and reliability, which are defined as follows:

$$Recognition\ Rate = \frac{N_{RECOG}}{N_{TESTED}} \times 100, \quad (12)$$

$$Error\ Rate = \frac{N_{ERR}}{N_{TESTED}} \times 100, \quad (13)$$

TABLE 3
Word Recognition Rate and Recognition Time for the HRS Alone and Different Lexicon Sizes

Lexicon Size	Word Recognition Rate (%) TOP 1	TOP 5	TOP 10	Recognition Time (sec/word)
10	98.84	99.96	100	0.010
1,000	91.01	96.32	97.71	0.273
10,000	81.06	90.58	92.36	1.992
40,000	73.23	84.64	87.91	7.516
80,000	68.65	81.32	85.10	14.46

$$Rejection\ Rate = \frac{N_{REJ}}{N_{TEST}} \times 100, \quad (14)$$

$$Reliability = \frac{N_{RECOG}}{N_{RECOG} + N_{ERR}} \times 100, \quad (15)$$

where N_{RECOG} is defined as the number of words correctly classified, N_{ERR} is defined as the number of words misclassified, N_{REJ} is defined as the number of input words rejected after classification, and N_{TESTED} is the number of input words tested.

The recognition time, defined as the time in seconds required to recognize one word and measured in CPU-seconds, was obtained on a PC Athlon 1.1GHz with 512MB of RAM memory running Linux. The recognition times are averaged over the test set.

5.1 Performance of the Handwriting Recognition System (HRS)

The performance of the HRS was evaluated on a very-large vocabulary of 85,092 city names, where the words have an average length of 11.2 characters. Table 3 shows the word recognition rate and processing time achieved by the HRS on five different dynamically generated lexicons. In Table 3, *TOP* 1, *TOP* 5, and *TOP* 10 denote that the truth word is the best word hypothesis or it is among the five best or the 10-best word hypotheses, respectively.

A common behavior of handwriting recognition systems which is also observed in Table 3 is the fall in recognition rate and the rise in recognition time as the size of the lexicon increases. Another important remark is the difference in recognition rate among the top one and the top 10-best word hypotheses. This indicates that the HRS is able to include among the top 10 word hypotheses the correct word with a recognition rate equal or higher than 85.10 percent, depending on the lexicon size.

Even though the recognition time presented in Table 3 might not meet the throughput requirements of many practical applications, it could be further reduced by using some code optimization, programming techniques, and parallel processing [7], [39].

5.2 Performance of the Segmental Neural Network (SNN)

The performance of the SNN was evaluated on a database of isolated handwritten characters derived from the SRTP database [33]. The training set contains 84,760 isolated characters with an equal distribution among the 52 classes (1,630 samples per character class). For those classes with low sample frequency, synthetic samples were generated by a stroke warping technique [33]. A validation set of 36,170 characters was also used during the training of the

TABLE 4
Character Recognition Rate on the SRTP and NIST Database Using the SNN as a Standard Classifier

Dataset	SRTP		NIST	
	Number of Samples	Recognition Rate (%)	Number of Samples	Recognition Rate (%)
Training	84,760	76.48	74,880	95.71
Validation	36,170	73.54	23,670	91.24
Test	46,700	73.51	23,941	87.79

SNN to monitor the generalization and a test set of 46,700 characters was used to evaluate the performance of the SNN.

Feature vectors composed of three feature types were generated from the character samples, as described in Section 3. The SNN was implemented according to the architecture described in Section 3 and it was trained using the backpropagation algorithm. The class probabilities at the output of the SNN were corrected to compensate for the changes in a priori class probabilities introduced by the frequency balancing.

Table 4 shows the character recognition rates achieved on the training, validation, and test set of the SRTP database when the SNN was used as a standard classifier. Notice that Table 4 also includes results on the training, validation, and test set of the NIST database. The recognition rates achieved by the SNN on the NIST database is similar to the performance of other character classifiers reported in the literature [33].

5.3 Performance of the Verification Stage (VS)

Several different test sets were generated from the *N*-best word hypothesis lists to assess the performance of the VS. Due to practical limitations, the number of experiments was restricted to five different lexicon sizes: 10, 1,000, 10,000, 40,000, and 80,000 words. The testing methodology for the verification of handwritten words is given below:

- Each word image in the test data set is submitted to the HRS and a list with the 10-best recognition hypotheses is generated. Such a list contains the ASCII transcription, the segmentation boundaries, and the a posteriori probability of each word hypothesis.
- Having the segmentation boundaries of each word hypothesis, we return to the word image to extract new features from each segment representing a character. The new feature vectors are used in the verification process.
- This procedure is repeated for each dynamic lexicon.

The performance of the VS alone is shown in Table 5. In this experiment, given the 10-best word hypotheses and segmentation boundaries generated by the HRS, the VS is invoked to produce confidence scores to each word hypothesis. The word recognition rate is computed based only on such scores. In Table 5, both the product (7) and the average rule (8) were considered to combine the character probabilities estimated by the SNN. Since the average rule has produced the best results, it was adopted for all further experiments.

There is a relative increase in recognition errors compared to the HRS for lexicons with less than 10,000 words (Table 3). On the other hand, for lexicons with 10,000, 40,000, and 80,000 words, the recognition rates achieved by the VS are better than those of the HRS. We attribute the worst performance on small lexicons to the presence of very

TABLE 5
Word Recognition Rates Considering Only the Confidence Scores Estimated by the VS to the *N*-Best Word Hypotheses, where the Confidence Scores Estimated by the SNN Are Combined by the Average (8) and Product (7) Rules

Lexicon Size	Word Recognition Rate (%)					
	Average			Product		
	TOP 1	TOP 5	TOP 10	TOP 1	TOP 5	TOP 10
10	94.04	98.76	100	82.08	83.45	100
1,000	88.21	95.84	97.71	78.31	81.67	82.01
10,000	82.53	91.12	92.36	73.38	77.86	78.07
40,000	75.97	85.98	87.91	67.81	73.53	73.94
80,000	72.39	83.09	85.10	65.15	71.32	71.71

different words in the word hypothesis list. Since the verification approach does not rely on the context but only averages character probabilities, it is more susceptible to errors. On the other hand, for large lexicons, the words that appear in the word hypothesis list are more similar in terms of both characters and lengths. In this case, the context is also "more similar" and, for this reason, it does not have a strong impact on the performance.

5.4 Performance of the Combination HRS+VS

The performance of the combination HRS and VS was evaluated by different combination rules. Table 6 shows the word recognition rate resulting from using the different rules to combine the confidence scores of the HRS and the VS for, 10 best word hypotheses and an 80,000-word lexicon. Higher recognition rates are achieved by using the weighted sum rule and the weighted product rule. Both weighted rules require setting up the weights to adjust the influence of each stage on the final composite confidence score. This was done using a validation data set and better performance was achieved using weights close to 0.15 and 0.85 for the HRS and VS, respectively. The *Max* rule is also an interesting combination rule because it provides results very close to the weighted rules and it does not require any adjustments of the parameters.

The results of combining the HRS and the VS by the weighted sum and the HRS alone are shown in Fig. 5. There is a significant improvement of almost 9 percent in recognition rate relative to the HRS alone. More moderate improvements were achieved for smaller lexicon sizes. Notice that the effects of the VS are gradually reduced as the size of the lexicon decreases, but it is still able to improve the recognition rate by about 0.5 percent for a 10-word lexicon. Fig. 5 also shows that the weighted sum is the best combination rule for all lexicon sizes. Finally, Table 7 summarizes the results achieved on different lexicon sizes by the HRS alone, the VS alone, and by the combination of both using the weighted sum rule.

TABLE 6
Word Recognition Rate for Different Combination Rules and an 80,000-Word Lexicon

Combination Rule	Word Recognition Rate (%)		
	TOP 1	TOP 2	TOP 5
Sum	74.52	81.77	84.59
Weighted Sum	77.62	81.79	84.27
Max	77.00	82.11	84.34
Product	70.65	77.17	82.93
Weighted Product	77.38	81.79	84.15

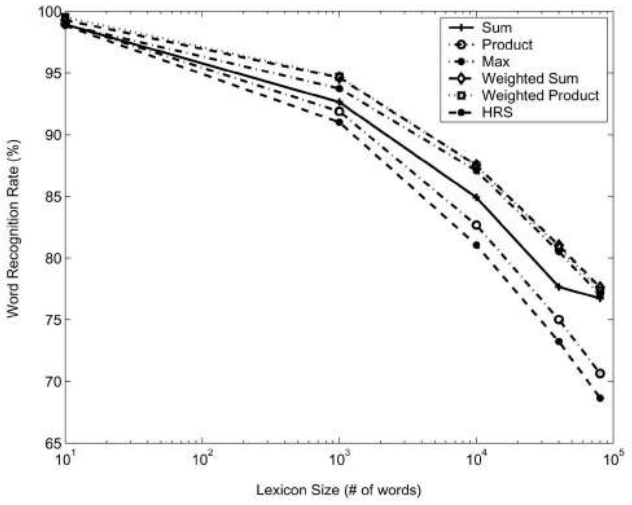


Fig. 5. Word recognition rate on different lexicon sizes for the HRS alone and for the combination of the HRS and the VS by different rules.

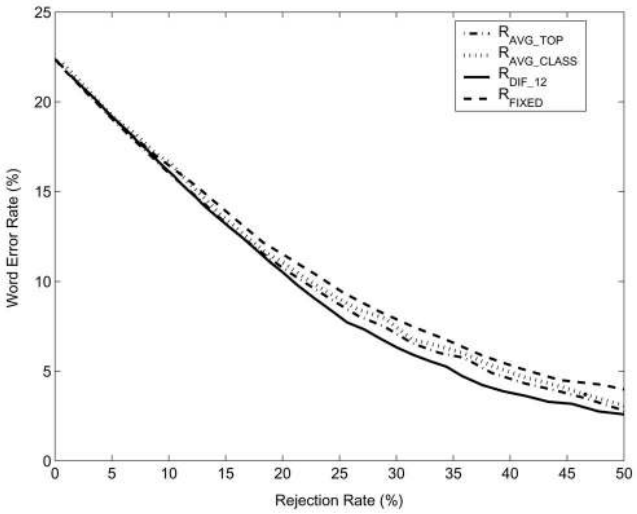


Fig. 6. Word error rate versus the rejection rate versus different rejection rules for an 80,000-word lexicon.

5.4.1 Error Analysis

In spite of the improvements in recognition rate brought about by a combination of the HRS and the VS, there is still a significant difference between the recognition rates of the top one and top 10 word hypotheses. This difference ranges from 0.71 percent to 7.48 percent for a 10-word and an 80,000-word lexicon, respectively. To better understand the role of the VS in the recognition of unconstrained handwritten words, we analyze the situations where the verifier succeeds in rescoring and reranking the truth word hypothesis (shifting it up to the top of the N-best word hypothesis list).

In the test set, 1,136 out of 4,674 words (24.31 percent) were reranked, where 488 words were correctly reranked (10.45 percent), that is, the truth word hypothesis was shifted up to the first position of the 10-best word hypothesis list and 648 words (13.86 percent) were not correctly reranked. However, for 446 out of 648 words (9.56 percent), the truth word hypothesis was not among the 10-best word hypotheses and, for the remaining 183 words (3.91 percent), 69 words were correctly recognized by the HRS alone (1.48 percent), but, after the combination with the VS, they were shifted down from the top of the list. For the remaining 114 out of 183 words (2.44 percent), the combination with the VS was not able to shift the true word hypothesis up to the first position of the list.

In summary, the combination of the HRS and the VS was able to correctly rerank 10.45 percent of the word hypotheses, shifting them up to the top of the list, but it also wrongly reranked 1.48 percent of the word hypotheses, shifting them down from the top of the list. This represents an overall improvement of 8.97 percent in the word recognition rate for an 80,000-word lexicon. Considering that the upper bound for improvement is the difference in recognition rate between the top one and the top 10, that is, 17 percent, and that the truth word hypothesis was not present in 9.56 percent of the 10-best word hypothesis lists, the improvement brought about by the combination of the HRS and the VS is very significant (more than 50 percent). The error analyses on other sizes of lexicons were also carried out and the proportion of the errors found was similar to those presented above.

TABLE 7
Word Recognition Rate for the HRS Alone, the VS Alone, and the Combination of Both by the Weighted Sum Rule (HRS + VS)

Lexicon Size	Approach	Word Recognition Rate (%)			
		TOP 1	TOP 2	TOP 5	TOP 10
10	HRS	98.84	99.74	99.96	100.00
	VS	94.04	96.31	98.76	—
	HRS+VS	99.29	99.78	99.98	—
1,000	HRS	91.01	94.20	96.32	97.71
	VS	88.21	92.65	95.84	—
	HRS+VS	94.63	96.23	97.30	—
10,000	HRS	81.06	85.83	90.58	92.36
	VS	82.53	87.05	91.12	—
	HRS+VS	87.53	90.52	92.32	—
40,000	HRS	73.23	79.48	84.64	87.91
	VS	75.97	81.46	85.98	—
	HRS+VS	81.02	84.79	87.18	—
80,000	HRS	68.65	75.50	81.32	85.10
	VS	72.39	78.63	83.09	—
	HRS+VS	77.62	81.79	84.27	—

5.5 Performance of the Decision Stage

We have applied the rejection criteria at the composite confidence scores produced by combining the outputs of the HRS and the VS to reject or accept the best word hypothesis. Fig. 6 shows the word error rate on the test data set as a function of rejection rate for different rejection criteria and considering an 80,000-word lexicon. Among the different rejection criteria, the one based on the difference between the confidence scores of the first best word hypothesis (H_1) and the second best word hypothesis (H_2) performs the best. A similar performance was observed for different lexicon sizes; however, it is more significant on large lexicons. Therefore, for all other experiments, we have adopted the R_{DIF_12} as the rejection criterion.

Fig. 7 shows the word error rates on the test data set as a function of rejection rate for a combination of the HRS and the VS for different lexicon sizes and using the R_{DIF_12} rejection criterion. If we compare such curves with those in Fig. 8 which were obtained by applying the same rejection criterion at the output of the HRS alone, it is clear the reduction in word error rate afforded by the combination of the HRS and the VS for the same rejection rates. For instance, at a 30 percent rejection

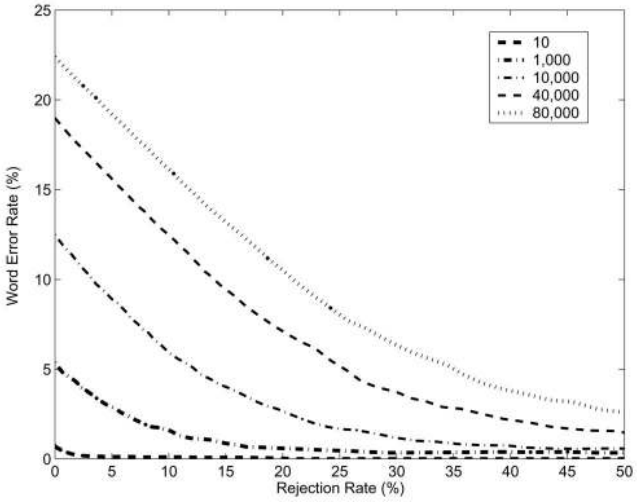


Fig. 7. Word error rate versus the rejection rate for the combination HRS+VS and different lexicon sizes.

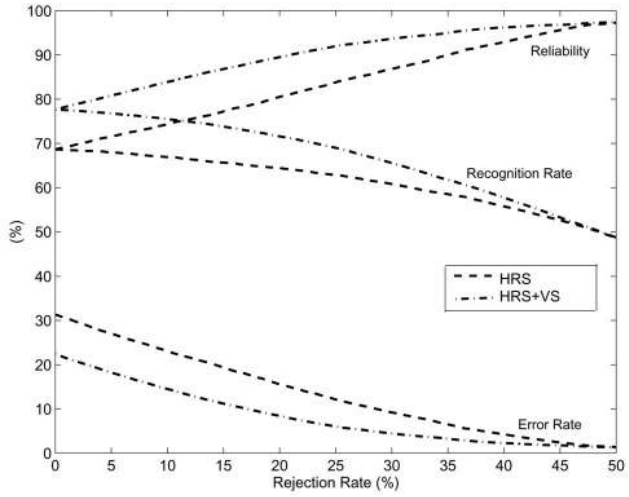


Fig. 9. Recognition rate, error rate, and reliability as a function of rejection rate for the HRS and for the combination HRS+VS and an 80,000-word lexicon.

level, the word error rate for the HRS is about 14 percent, while, for the combination, HRS and VS is about 6 percent for an 80,000-word lexicon. A similar behavior is observed for other lexicon sizes and rejection rates.

Finally, the last aspect that is interesting to analyze is the improvement in reliability afforded by the VS. Fig. 9 shows the evolution of the recognition rate, error rate, and reliability as a function of the rejection rate. We can observe that, for low rejection rates, a combination of the HRS and the VS produces interesting error-reject trade-off compared to the HRS alone.

5.6 Evaluation of the Overall Recognition and Verification System

We have not given much attention to the recognition time until now. Nevertheless, this important aspect may diminish the usability of the recognition system in practical applications that deal with large and very large vocabularies. A lot of effort has been devoted to building a fast handwriting recognition system [29], [33]. Therefore, at this point, it is worthwhile to ascertain the impact of the VS on the whole recognition process, that is, in both the recognition rate and the recognition time. Table 8 shows the computation break down for the VS where the results depend on the number of word hypotheses provided by the HRS as well as on the length of such hypotheses (number of characters). The results shown in Table 8 are for a list of 10 word hypotheses, where the words have an average of 11 characters. Besides that, the feature extraction step also depends on the number of pixels that represent the input characters.

To draw any conclusion, it is also necessary to know the time spent by the HRS to generate an N-best word hypothesis list. Table 9 shows the computation breakdown for the HRS. Notice that, in this case, preprocessing, segmentation, and feature extraction steps depend on the word length and number of pixels of the input, image while the recognition step depends on the lexicon size. By comparing Tables 8 and 9, it is possible to ascertain that the time required by the verification process corresponds to less than 1 percent of the time required by the HRS to generate a list of 10 word hypotheses. However, it can be argued that the nearly

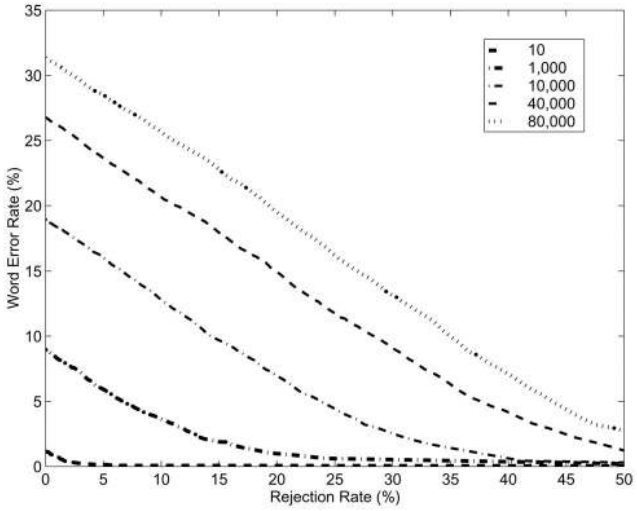


Fig. 8. Word error rate versus the rejection rate for the HRS alone and different lexicon sizes.

TABLE 8
Computation Breakdown for the Verification of Handwritten Words for a List of 10 Word Hypotheses and an 80,000-Word Lexicon

Step	Average Time (msec/word)
Feature Extraction	48
Recognition	4
Combination	2
Total	60

TABLE 9
Computation Breakdown for the Recognition of Handwritten Words for 10 and 80,000-Word Lexicons

Step	Average Time (sec/word)	
	10-word lexicon	80,000-word lexicon
Pre-processing	355m	
Segmentation	110m	
Feature Extraction	34m	
Recognition	10m	15
Total	510m	16

9 percent rise in the word recognition rate afforded by the VS is worthwhile.

The time required for verification does not depend on the lexicon size, but only on the number of word hypotheses provided by the HRS as well as on the number of characters in each word hypothesis. On the other hand, the time spent in recognition by the HRS strongly depends on the lexicon size as well as on the number of characters in each word hypothesis. For a 10-word lexicon, the overall recognition time is approximately one second. Therefore, the verification process now corresponds to about 13 percent of the time required by the HRS to generate a list of the 10-best word hypotheses. It can be argued that the nearly 0.5 percent rise in the word recognition rate afforded by the VS is still useful.

6 Conclusion

In this paper, we have presented a novel verification approach that relies on the strengths and knowledge of weaknesses of an HMM-based recognition system to improve its performance in terms of recognition rate while not increasing the recognition time. The proposed approach combines two different classification strategies that operate in different representation spaces (word and character). The recognition rate resulting from a combination of the recognition and verification approach is significantly better than that achieved by the HMM-based recognition system alone, while the delay introduced in the overall recognition process is almost negligible. This last remark is very important, especially when tackling very-large vocabulary recognition tasks, where recognition speed is an issue as important as recognition rate. For instance, on an 80,000-word lexicon, the HMM-based recognition system alone achieves recognition rates of about 68 percent. Using the verification strategy, it is possible to achieve recognition rates of about 78 percent with 1 percent delay in the overall recognition process. At the 30 percent rejection level, the reliability achieved by the combination of the recognition and verification approaches is about 94 percent. Nevertheless, the improvement in performance is also advantageous for small and medium vocabulary recognition tasks.

Compared with previous works on the same data set [4], [9], [10], [28], [36], the results reported in this paper represent a significant improvement in terms of recognition rate, reliability, and recognition time. It is very difficult to compare the performance of the proposed approach with other results available in the literature due to the differences in experimental conditions and particularly because we have considered unconstrained handwritten words and very large vocabularies. Recent works in large vocabulary handwriting recognition report recognition rates between 80 percent and 90 percent. Arica and Yarman-Vural [40] have achieved 88.8 percent recognition rate for a 40,000-word vocabulary on a single author test set of 2,000 cursive handwritten words. Senior and Robinson [2] have achieved 88.3 percent recognition rate on the same data set with a lexicon of size 30,000. Carbonnel and Anquetil [41] have achieved 80.6 percent recognition rate on a test set of 2,000 handwritten words. Vinciarelli et al. [8] have achieved 46 percent accuracy on the recognition of handwritten texts. Other results on large vocabulary handwriting recognition are presented in [36]. The aspect of recognition speed is much more difficult to compare with other works since recognition time is not reported by most of the authors. Even considering such difficulties, the results reported in this paper are very relevant since they were obtained on a large data set and the word images were extracted from real postal envelopes.

In spite of the good results achieved, there are some shortcomings related to the verification approach. The first shortcoming is that verification depends on the output of the HMM-based recognition system. If the truth word hypothesis is not present in the N-best word hypothesis list, the verification becomes useless. However, this problem can be alleviated by using a great number of word hypotheses. It is clear that the more word hypotheses we take, the higher the recognition rate becomes [33]. However, in the scope of this paper, it would be impractical to consider a higher number of word hypotheses. Automatic selection of the number of word hypotheses based on the a posteriori probabilities may help to improve the recognition rate by selecting more word hypotheses when necessary. Another shortcoming is the assumption that the segmentation of words into characters carried out by the HMM-based recognition system is reliable. However, about 20 percent of the words are wrongly segmented. A postprocessing of the segmentation points at the verification level could be useful to overcome the segmentation problem and it could help to boost the improvements brought about by the verification stage. Both topics will be the subject of future research.

From the experimental point of view, another shortcoming is the poor quality of the data used for training the segmental neural network. Nevertheless, even with this limitation, the use of the segmental neural network to estimate probabilities to the segmentation hypotheses provided by the HMM-based recognition system succeeded very well and brought significant improvements in the word recognition rate. On the other hand, while the quality of the character samples in the training data set is not good, the data set was generated automatically by bootstrapping with no human intervention. This aspect is very relevant since gathering, segmenting, and labeling character by hand would be very tedious, time-consuming, and better results than those reported in this paper cannot be guaranteed.

The use of the rejection rules proposed in this paper has been shown to be a powerful method of reducing the error rate and improving the reliability. The results obtained by the proposed rejection rule applied over the confidence scores resulting from the combination of the HRS and the VS can be significantly improved over the stand-alone HRS. The reliability curves have shown significant gains through the use of the VS.

In summary, the main contribution of this paper is a novel approach that combines recognition and verification to enhance the word recognition rate. The proposed combination is effective and computationally efficient. Even if the verification stage is not fully optimized, the improvements reported in this paper are significant. Hence, it is logical to conclude that a combination of recognition and verification approaches is a promising research direction in handwriting recognition.

ACKNOWLEDGMENTS

The authors would like to acknowledge the CNPq-Brazil and the MEQ-Canada for the financial support and the SRTP-France for providing us with the database and an earlier version of the handwriting recognition system.

REFERENCES

[1] P.D. Gader, M.A. Mohamed, and J.M. Keller, "Fusion of Handwritten Word Classifiers," *Pattern Recognition Letters,* vol. 17, pp. 577-584, 1996.

[2] A.W. Senior and A.J. Robinson, "An Off-Line Cursive Handwriting Recognition System," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 3, pp. 309-321, Mar. 1998.

[3] T. Steinherz, E. Rivlin, and N. Intrator, "Offline Cursive Script Word Recognition—A Survey," *Int'l J. Document Analysis and Recognition,* vol. 2, pp. 90-110, 1999.

[4] A. El-Yacoubi, M. Gilloux, R. Sabourin, and C.Y. Suen, "Unconstrained Handwritten Word Recognition Using Hidden Markov Models," *IEEE Trans. Pattern Analysis and Machine Intelligence* vol. 21, no. 8, pp. 752-760, Aug. 1999.

[5] R. Plamondon and S.N. Srihari, "On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 1, pp. 68-89, Jan. 2000.

[6] N. Arica and F.T. Yarman-Vural, "An Overview of Character Recognition Focused on Off-Line Handwriting," *IEEE Trans. Systems, Man, and Cybernetics-Part C: Applications and Rev.,* vol. 31, no. 2, pp. 216-233, 2001.

[7] A.L. Koerich, R. Sabourin, and C.Y. Suen, "Large Vocabulary Off-Line Handwriting Recognition: A Survey," *Pattern Analysis and Applications,* vol. 6, no. 2, pp. 97-121, 2003.

[8] A. Vinciarelli, S. Bengio, and H. Bunke, "Offline Recognition of Unconstrained Handwriting Texts Using HMMs and Statistical Models," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 6, pp. 709-720, June 2004.

[9] C. Farouz, "Reconnaissance de Mots Manuscrits Hors-Ligne dans un Vocabulaire Ouvert par Modelisation Markovienne," PhD dissertation, Université de Nantes, Nantes, France, Aug. 1999.

[10] F. Grandidier, R. Sabourin, and C.Y. Suen, "Integration of Contextual Information in Handwriting Recognition Systems," *Proc. Seventh Int'l Conf. Document Analysis and Recognition,* pp. 1252-1256, 2003.

[11] R.K. Powalka, N. Sherkat, and R.J. Whitrow, "Word Shape Analysis for a Hybrid Recognition System," *Pattern Recognition,* vol. 30, no. 3, pp. 412-445, 1997.

[12] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas, "On Combining Classifiers," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, no. 3, pp. 226-239, Mar. 1998.

[13] C.Y. Suen and L. Lam, "Multiple Classifier Combination Methodologies for Different Output Levels," *Proc. First Int'l Workshop Multiple Classifier Systems,* pp. 52-66, 2000.

[14] S. Srihari, "A Survey of Sequential Combination of Word Recognizers in Handwritten Phrase Recognition at Cedar," *Proc. First Int'l Workshop Multiple Classifier Systems,* pp. 45-51, 2000.

[15] B. Verma, P. Gader, and W. Chen, "Fusion of Multiple Handwritten Word Recognition Techniques," *Pattern Recognition Letters,* vol. 22, pp. 991-998, 2001.

[16] L.S. Oliveira, R. Sabourin, F. Bortolozzi, and C.Y. Suen, "A Modular System to Recognize Numerical Amounts on Brazilian Bank Cheques," *Proc. Sixth Int'l Conf. Document Analysis and Recognition,* pp. 389-394, 2001.

[17] S. Madhvanath, E. Kleinberg, and V. Govindaraju, "Holistic Verification of Handwritten Phrases," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, no. 12, pp. 1344-1356, Dec. 1999.

[18] L.P. Cordella, P. Foggia, C. Sansone, F. Tortorella, and M. Vento, "A Cascaded Multiple Expert System for Verification," *Proc. First Int'l Workshop Multiple Classifier Systems,* pp. 330-339, 2000.

[19] S.J. Cho, J. Kim, and J.H. Kim, "Verification of Graphemes Using Neural Networks in an HMM-Based On-Line Korean Handwriting Recognition System," *Proc. Seventh Int'l Workshop Frontiers in Handwriting Recognition,* pp. 219-228, 2000.

[20] H. Takahashi and T.D. Griffin, "Recognition Enhancement by Linear Tournament Verification," *Proc. Int'l Conf. Document Analysis and Recognition,* pp. 585-588, 1993.

[21] A.S. Britto, R. Sabourin, F. Bortolozzi, and C.Y. Suen, "A Two-Stage HMM-Based System for Recognizing Handwritten Numeral Strings," *Proc. Sixth Int'l Conf. Document Analysis and Recognition,* pp. 396-400, 2001.

[22] D. Guillevic and C.Y. Suen, "Cursive Script Recognition Applied to the Processing of Bank Cheques," *Proc. Third Int'l Conf. Document Analysis and Recognition,* pp. 11-14, 1995.

[23] J. Gloger, A. Kaltenmeier, E. Mandler, and L. Andrews, "Reject Management in a Handwriting Recognition System," *Proc. Fourth Int'l Conf. Document Analysis and Recognition,* pp. 556-559, 1997.

[24] N. Gorski, "Optimizing Error-Reject Trade Off in Recognition Systems," *Proc. Fourth Int'l Conf. Document Analysis and Recognition,* pp. 1092-1096, 1997.

[25] S. Marukatat, T. Artieres, P. Gallinari, and B. Dorizzi, "Rejection Measures for Handwriting Sentence Recognition," *Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 24-29, 2002.

[26] J.F. Pitrelli and M.P. Perrone, "Confidence Modeling for Verification Post-Processing for Handwriting Recognition," *Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 30-35, 2002.

[27] A. Stolcke, Y. Konig, and M. Weintraub, "Explicit Word Error Minimization in N-Best List Rescoring," *Proc. Eurospeech '97,* pp. 163-166, 1997.

[28] A.L. Koerich, R. Sabourin, and C.Y. Suen, "Fast Two-Level Viterbi Search Algorithm for Unconstrained Handwriting Recognition," *Proc. 27th Int'l Conf. Acoustics, Speech, and Signal Processing,* pp. 3537-3540, 2002.

[29] A.L. Koerich, R. Sabourin, and C.Y. Suen, "Fast Two-Level HMM Decoding Algorithm for Large Vocabulary Handwriting Recognition," *Proc. Ninth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 232-237, 2004.

[30] G. Zavaliagkos, Y. Zhao, R. Schwartz, and J. Makhoul, "A Hybrid Segmental Neural Net/Hidden Markov Model System for Continuous Speech Recognition," *IEEE Trans. Speech and Audio Processing,* vol. 2, no. 1, pp. 151-160, 1994.

[31] F. Grandidier, "Analyse de l'Ensemble des Confusions d'un Système," technical report, École de Technologie Supérieure, Montréal, Canada, 2000.

[32] H. Duplantier, "Interfaces de Visualisation pour la Reconnaissance d'Écriture," technical report, École de Technologie Supérieure, Montréal, Canada, 1998.

[33] A.L. Koerich, "Large Vocabulary Off-Line Handwritten Word Recognition," PhD dissertation, École de Technologie Supérieure de l'Université du Québec, Montréal, Canada, 2002.

[34] L. Yaeger, R. Lyon, and B. Webb, "Effective Training of a Neural Network Character Classifier for Word Recognition," *Proc. Advances in Neural Information Processing Systems,* pp. 807-813, 1997.

[35] A.L. Koerich, Y. Leydier, R. Sabourin, and C.Y. Suen, "A Hybrid Large Vocabulary Handwritten Word Recognition System Using Neural Networks with Hidden Markov Models," *Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 99-104, 2002.

[36] A.L. Koerich, R. Sabourin, and C.Y. Suen, "Lexicon-Driven HMM Decoding for Large Vocabulary Handwriting Recognition with Multiple Character Models," *Int'l J. Document Analysis and Recognition,* vol. 6, no. 2, pp. 126-144, 2003.

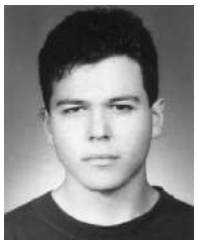
[37] J. Schurmann, *Pattern Classification: A Unified View of Statistical and Neural Approaches.* John Wiley and Sons, 1996.

[38] A.L. Koerich, "Rejection Strategies for Handwritten Word Recognition," *Proc. Ninth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 479-484, 2004.

[39] A.L. Koerich, R. Sabourin, and C.Y. Suen, "A Distributed Scheme for Lexicon-Driven Handwritten Word Recognition and Its Application to Large Vocabulary Problems," *Proc. Sixth Int'l Conf. Document Analysis and Recognition,* pp. 660-664, 2001.

[40] N. Arica and F.T. Yarman-Vural, "Optical Character Recognition for Cursive Handwriting," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 6, pp. 801-813, June 2002.

[41] S. Carbonnel and E. Anquetil, "Lexical Post-Processing Optimization for Handwritten Word Recognition," *Proc. Seventh Int'l Conf. Document Analysis and Recognition,* pp. 477-481, 2003.



Alessandro L. Koerich received the BSc degree in electrical engineering from the Federal University of Santa Catarina (UFSC), Brazil, in 1995, the MSc degree in electrical engineering from the University of Campinas (UNICAMP), Brazil, in 1997, and the PhD degree in automated manufacturing engineering from the École de Technologie Supérieure, Université du Québec, Montréal, Canada, in 2002. From 1997 to 1998, he was a lecturer at the Federal Center for Technological Education (CEFETPR). From 1998 to 2002, he was a visiting scientist at the Centre for Pattern Recognition and Machine Intelligence (CENPARMI). In 2003, he joined the Pontifical Catholic University of Paraná (PUCPR), Curitiba, Brazil, where he is currently an associate professor of computer science. He is cofounder of INVISYS, a R&D company that develops machine vision systems. In 2004, Dr. Koerich was nominated an IEEE CS Latin America Distinguished Speaker. He is member of the Brazilian Computer Society, IEEE, IAPR, and ACM. He is the author of more than 50 papers and holds patents in image processing. His research interests include machine learning, machine vision, and multimedia.

Robert Sabourin received the BIng, MScA, and PhD degrees in electrical engineering from the École Polytechnique de Montréal in 1977, 1980, and 1991, respectively. In 1997, he joined the Physics Department of Montréal University, where he was responsible for the design, experimentation, and development of scientific instrumentation for the Mont Megantic Astronomical Observatory. His main contribution was the design and the implementation of a microprocessor-based fine tracking system combined with a low-light-level CCD detector. In 1983, he joined the staff of the École de Technologie Supérieure, Université du Québec, in Montréal, where he cofounded the Department of Automated Manufacturing Engineering where he is currently a full professor and teaches pattern recognition, evolutionary algorithms, neural networks, and fuzzy systems. In 1992, he also joined the Computer Science Department of the Pontifical Catholic University of Paraná (Curitiba, Brazil) where, in 1995, he was co-responsible for the implementation of a masters program and, in 1998, a PhD program in applied computer science. Since 1996, he has been a senior member of the Centre for Pattern Recognition and Machine Intelligence (CENPARMI). Dr Sabourin is the author (and coauthor) of more than 150 scientific publications, including journals and conference proceedings. He was cochair of the program committee of CIFED '98 (Conférence Internationale Francophone sur l'Écrit et le Document, Québec, Canada) and IWFHR '04 (Ninth International Workshop on Frontiers in Handwriting Recognition, Tokyo, Japan). He was nominated as conference cochair of the next ICDAR '07 (Ninth International Conference on Document Analysis and Recognition) that will be held in Curitiba, Brazil, in 2007. His research interests are in the areas of handwriting recognition and signature verification for banking and postal applications. He is a member of the IEEE.

Ching Y. Suen received the MSc (eng.) degree from the University of Hong Kong and the PhD degree from the University of British Columbia, Canada. In 1972, he joined the Department of Computer Science of Concordia University, Montréal, Canada, where he became a professor in 1979 and served as chairman from 1980 to 1984, and as associate dean for research of the Faculty of Engineering and Computer Science from 1993 to 1997. He has guided/hosted 65 visiting scientists and professors and supervised 60 doctoral and master's graduates. Currently, he holds the distinguished Concordia Research Chair in Artificial Intelligence and Pattern Recognition, and is the Director of CENPARMI, the Centre for PR & MI. Professor Suen is the author/editor of 11 books and more than 400 papers on subjects ranging from computer vision and handwriting recognition, to expert systems and computational linguistics. A Google search of "Ching Y. Suen" will show some of his publications. He is the founder of the *International Journal of Computer Processing of Oriental Languages* and served as its first editor-in-chief for 10 years. Presently, he is an associate editor of several journals related to pattern recognition. He is a fellow of the IEEE, IAPR, and the Academy of Sciences of the Royal Society of Canada and he has served several professional societies as president, vice-president, or governor. He is also the founder and chair of several conference series including ICDAR, IWFHR, and VI. He had been the general chair of numerous international conferences, including the International Conference on Computer Processing of Chinese and Oriental Languages in August 1988 held in Toronto, International Conference on Document Analysis and Recognition held in Montréal in August 1995, and the International Conference on Pattern Recognition held in Québec City in August 2002. Dr. Suen has given 150 seminars at major computer industries and various government and academic institutions around the world. He has been the principal investigator of 25 industrial/government research contracts and is a grant holder and recipient of prestigious awards, including the ITAC/NSERC Award from the Information Technology Association of Canada and the Natural Sciences and Engineering Research Council of Canada in 1992 and the Concordia "Research Fellow" award in 1998, and the IAPR ICDAR award in 2005.