Recognition of Handwritten Dates on Bankchecks Using an HMM Approach

MARISA EMIKA MORITA[1], EDOUARD LETHELIER[1], ABDENAIM EL YACOUBI[1],
FLÁVIO BORTOLOZZI[1], ROBERT SABOURIN[2]

[1]PUCPR - Pontifícia Universidade Católica do Paraná
PPGIA - Programa de Pós-Graduação em Informática Aplicada
LARDOC - Laboratório de Análise e Reconhecimento de Documentos
Rua Imaculada Conceição 1155, Prado Velho, 80215-901 - Curitiba - Pr
{marisa,edouard,yacoubi,fborto}@ppgia.pucpr.br

[2]ETS - Ecole de Technologie Supérieure
LIVIA - Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle
1100, rue Notre Dame Ouest, Montréal, H3C 1K3, Canada
sabourin@gpa.etsmtl.ca

Abstract. This article presents the first results of our system applied to recognize automatically the handwritten dates on Brazilian bankchecks. Considering the omni-writer context, we detail here our recognition module dedicated to process the month field. This module is based on the combination of holistic and analytical approaches with a fixed lexicon. Both approaches operate with a single explicit segmentation technique to provide a grapheme sequence for the purposed Hidden Markov Models of each recognizer. We show the significative improvements in combining both modules to get a satisfactory recognition rate considering the small database images we work with. Finally, we present various perspectives for our future works.

1 Introduction

Various works and applications have been carried out in the field of off-line handwriting recognition to try to reach the human behavior while reading handwritten words. Despite the high computing capabilities of current computers, these systems are usually unable to correctly replicate tasks as the human vision. Therefore the production in this area is still intensive.

In handwriting recognition, most of obstacles remain in the strong variability of the handwriting styles for omni-writer context, and also in the size of the vocabulary (lexicon size). Moreover, the off-line approach is more complex than the on-line approach, due to the noisy effects in the image acquisition process and the lost of temporal information such as the writing sequence and the velocity, very helpful in a recognition process.

In order to reduce the complexity of the problem, many research teams use contextual information in their systems by adopting a lexicon-based strategy. In applications where the lexicon size is adequate, e.g in the context of the legal amount on checks, a holistic approach can be considered to provide a specific word model for each determined word class [5], [2]. However, when the lexicon size is large, e.g in the postal context of city names, most of current techniques are based on analytical approaches. In such a case, one model for each character class is built. Then, the word models are built by the concatenation of the appropriated character models [4], [5], [6], [8].

In analytical approach, the words are segmented into graphemes that can represent characters or pseudo-characters. This segmentation strategy is the most used, due to the large variability of the handwriting. In this case, the definitive segmentation points are determined in the recognition phase by the graphemes concatenation. Even if the holistic approach does not require any kind of segmentation, the analytical approach has advantages over the holistic approach. Once the character alphabet is learned by the analytical models, any context containing these characters can be considered by the system. This portability is very useful when the document image contains various fields to be processed by the same word recognizer. Moreover, the training of the characters does not depend of the lexicon size.

The Hidden Markov Models (HMMs) have been successfully applied in speech recognition. More recently, they have been used in handwritten words recognition [4], [5], [6], [8], [1]. The HMMs word architecture is pretty well adapted to describe a word image as a sequence of observations. Some approaches are based on explicit segmentation where words are cut into characters or pseudo-characters to provide the grapheme sequence [5], [8], [1]. In some holistic approaches, the explicit segmentation can be also used to improve the word-length estimation (for a given word-class) and to permit a better class discrimination

0-7695-0878-2/00 $10.00 © 2000 IEEE

while training the word models. In implicit segmentation approach, the cut hypotheses are implicitly contained in the feature description of sequential vertical frames provided to the HMMs [4], [6].

The aim of our work is focused on the off-line recognition of handwritten dates on brazilian bankchecks, considering an omni-writer context. The peculiarity of our application is based on the complexity to process as a hole the mixed information composed by the date field: words (city, month, separator) and digits (day and year). We show in this article the first results of our works dedicated to the recognition of isolated month words. Although this study deals with a limited lexicon size (12 classes), we try to improve the discrimination between specific classes when they contain a common sub-string as the classes "Setembro" (September) and "Novembro" (November). To face this problem, we developed a combination strategy involving the both holistic and analytical HMMs approaches using explicit segmentation technique. For the training and the test of the HMMs models, we use our laboratory database which contains about 2,000 images of extracted fields of brazilian bankchecks with a resolution of 300 dpi.

We describe in section 2 the context of dates in brazilian bankchecks. In section 3, we detail our segmentation approach to provide the sequence of observations of the models. In section 4, we describe our classification approaches and we show the experimentation results in section 5. Finally, the section 6 presents our conclusions and perspectives in our future works.

2 Peculiarities of our study

There has been a lot of work dedicated to the processing of literal handwritten amounts on checks. Nevertheless, the studies about the date recognition are fewer, even inexistent for the brazilian type.

For almost countries, the date field of a check must be filled to be validated. If not, a brazilian check must be returned to its owner. The date information consists of the following fields, presented below as they appear from left to right:

- city name (alphabetical) optional;
- **coma** (separator);
- day (numerical);
- "**de**" (separator);
- month (alphabetical);
- "**de**" (separator);
- year (numerical).

If one of the data format is not respected, the check will not be validated. The three field separators (in bold) are usually printed on the check as well as the baseline. Figure 1 shows an example of our laboratory database where we do not consider any extraction process of the date from the background of check.

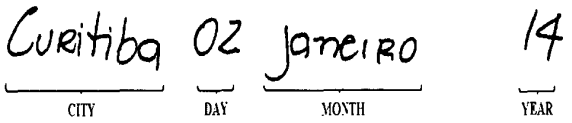

Figure 1: Example of our laboratory database image

Even if the field separators were not pre-printed before collecting our database, we have observed various images where writers put their own separators while filling the date, such as the coma (,), (;), point (.) or "de". Table 1 contains the whole vocabulary of our laboratory database for the three obligatory parts to be filled (e.g day, month and year) and its variabilities, such as the two-digits day (01...09) and 2/4 digits year (1999 or 99).

Table 1: Vocabulary of the date field in our database

Day	Month	Year
(0)1	Janeiro	(19)97
(0)2	Fevereiro	(19)98
.	Março	.
.	Abril	.
.	Maio	.
10	Junho	(20)00
.	Julho	.
.	Agosto	.
20	Setembro	(20)10
.	Outubro	.
.	Novembro	.
31	Dezembro	(20)20

3 Isolated word representation

Our current recognition system only deals with isolated words corresponding to the month field. After being manually located on the date image, the month word is explicitly segmented into graphemes, which are then converted into discrete symbols after the feature extraction stage. The resulting sequence of symbols is considered as the input to our HMM model. We employ in this work global features (loops, ascenders and descenders).

Figure 2: Significant spaces detection

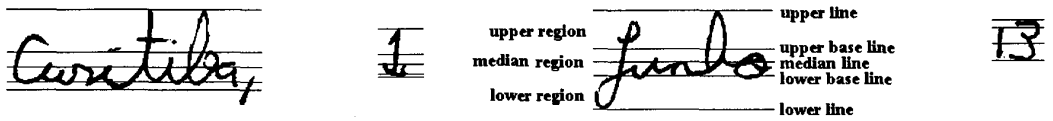

Figure 3: Reference lines detection

3.1 Date segmentation

The goal of this process is to try to localize each relevant field in the date (cf. section 2). This algorithm is based on space detection between connected components (*CCs*). An averaged space length is computed from all the detected spaces in the median region of the date. We use the transition histogram to determine the median region. Based on this averaged length, we compute a threshold to localize every significant space between the entities. Even if all the entities are not correctly segmented (as shown in Figure 2 with the city name and the coma), this very simple approach is sufficient to yield a proper estimation of the reference lines we express from the sub-images of the date.

3.2 Reference lines detection

Considering a cursive word, we can define the reference baselines as the vertical transitions between the upper and the median regions and between the lower and the median regions (see Figure 3). Commonly, the median region contains all the lower-case letters of the word, and the upper and lower regions contain all the letter extensions.

The baseline detection is applied for each segmented sub-image of the date. The upper (lower) baseline is provided by the average height of the upper contour maxima (lower contour minima) previously filtered using the weighted least square technique. The upper and lower lines correspond to the highest point of the upper contour and to the lowest point of the lower contour respectively. The median line is the half distance between the upper and lower baselines.

3.3 Character segmentation of words

The main goal of the character segmentation is to provide a sequence of observations or graphemes from a given word. In analytical approach, this sequence is compared (and evaluated) to the combination of the character models for each word of the lexicon. In some cases, the segmentation technique could be also encountered in holistic approaches to improve the word length estimation and to facilitate the primitive detection. In our study, we worked with isolated month words.

The detection of the segmentation points (*SPs*) is based on two hypotheses:

- Some characters are naturally isolated in the word (letter "n" in Figure 4);
- The local minima of the upper contour of *CCs* (*MPs*) correspond to ligatures between characters.

The *SPs* hypotheses are validated if they belong to a determined segmentation height H_s, delimited by the upper and lower limits (Figure 4(a)). These limits represent 40% of the median region height when the lower and upper lines are greater than this value. Thus, in Figure 4(a), the upper limit corresponds to the upper line and the lower limit corresponds 40% of the median region height. A new area is then determined as the significative lower region to detect some hypothetical descenders.

The generation of *SPs* results from the validation of the *MPs*. This validation is based on the detection of the lower contour on the vertical projection of the *MP*. In particular cases, the *SP* is shifted from the original *MP* location. It occurs when:

- The vertical projection crosses a loop before reaching the lower contour (Figure 5(a));
- The vertical projection remains tangent with the lower contour (Figure 5(b));
- The width of the vertical projection is not acceptable (Figure 5(c)).

For any of these three cases, the algorithm processes a right shift from the *MP* location, following the upper contour, to try to find a new *SP* to validate. A *MP* corresponds to a *SP* if it can be shifted in its right neighborhood at an upper contour point minimizing the vertical width with respect to the lower contour without crossing a loop or a tangent. Otherwise, the algorithm tries the same search in the left neighborhood of the *MP*.

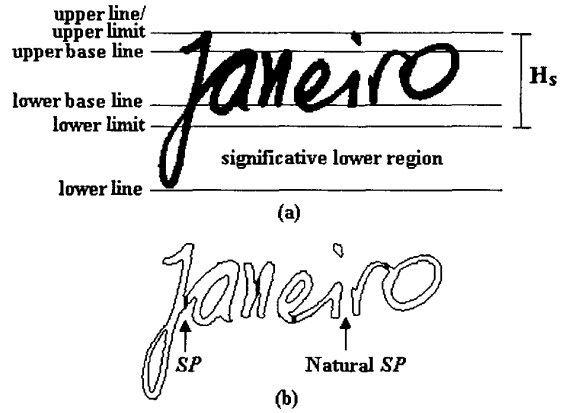

Figure 4: Segmentation points (a): delimitation of the segmentation region, (b): types of *SPs*

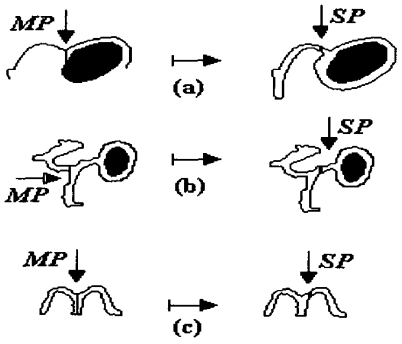

Figure 5: Cases of shifted *SPs* (a): with loop, (b): contour tangency, (c): inacceptable *SP* width

For two particular cases, any validated *SP* can be removed from the word image:

- The contour distance (number of adjacent pixels of contour) between two starts of *SP* is lower than a threshold T_s, proportional to the stroke thickness;
- The contour distance between the *SP* and the right or left limit of the *CC* is lower than T_s.

After running our segmentation algorithm over the image database, we observed that the system may produce a correct word segmentation in characters (ideal), an under-segmentation (at least two characters remain linked) or an over-segmentation where a character contains two or three graphemes. In section 4, we detail the topology of our recognition model according to these observations.

3.4 Primitive extraction

To face the difficult problem of selecting the best primitive set to process the month words, we first decided to implement and test one of the most popular type of primitives opted and validated in various holistic approaches, namely ascenders, descenders and loops.

The primitive extraction algorithm is based on the detection of local maxima from the upper contour and local minima from the lower contour. Depending on the region where each extremum is located, the extension or loop is classified as a big or short primitive.

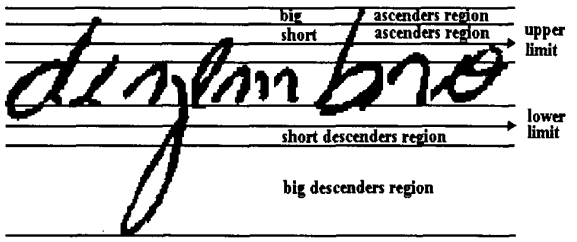

Figure 6: Regions for ascenders and descenders detection

To improve the discrimination of primitives and after various trials, the significant upper and lower regions are splitted into two sub-regions:

- A short region, corresponding to 40% of the median region height;
- A big region, covering the complementarity of the significant region.

To avoid spurious detections (as false ascenders or descenders) in the neighborhood of the median region, we also defined a neutral region between the short and the median regions (upper and lower limits in Figure 6). The gap of the neutral regions is also 40% of the median region height.

Thus, for each grapheme, the detection is processed as follow:

- If a local maximum is located in the short upper region, it corresponds to a short ascender;
- If a local maximum is located in the big upper region, it corresponds to a big ascender;

- If a local minimum is located in the short lower region, it corresponds to a short descender;
- If a local minimum is located in the big lower region, it corresponds to a big descender.

When belonging to the median region, we classify the loop in two categories (big or short) of primitives depending on its vertical size. When a loop is located in the upper or in the lower region, it is implicitly associated with an ascender or a descender. But conversely to the primitive-set proposed by El Yacoubi et al. [8], our primitive extraction does not provide any specific class taking into account the detection order to be able to discriminate characters as "b"/"d" and "p"/"q". Indeed, the month field does not contain any "p" nor "q" and the positions of characters "b" and "d" in the month words are different.

Finally, to improve the words discrimination in our recognition system, we also use a specific primitive class ("-") when a grapheme does not contain any relevant primitive. This class considerably improves the discrimination through the word length estimation. Then, we obtain a 20 symbols alphabet where each symbol has been evaluated on the training set in order to provide a significant consistency to the HMMs.

4 HMM-based isolated word recognition using a limited lexicon

Due to its limited lexicon size (12 word classes), the context of isolated month processing seems to be naturally well adapted to the holistic approach. However, in handwriting recognition, the performance of the system strongly depends on the size of the training set. Since we started working with a small database, we decided to implement both holistic and analytical approaches. The motivation is duple: considering the analytical approach, we can significantly increase the size of the database from the same data set and we can also evaluate the veracity of our primitive set through both recognition systems based on HMMs.

4.1 HMMs topology

For our both approaches, we use the left-right model (Bakis) in order to consider the writing arrangement of characters. The observation sequences are emitted on the model transitions in order to take advantage of the explicit segmentation output. Figure 7 details the character model we have chosen, according to the definitions met in [8]. This architecture is able to consider the various configurations of the segmentation. Thus, the transition $t_{03} = \Phi$ models character under-segmentation. The transitions t_{01}, t_{12} and t_{23} model character segmentation into 2 or 3 graphemes. This topology permits a better absorption in the homogeneity of the graphemes provided by the segmentation. Considering upper-case and lower-case characters, our lexicon contains 40 HMMs. Since the month alphabet is reduced to 20 character classes, we do not consider unused characters.

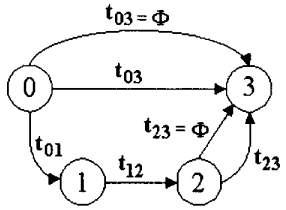

Figure 7: Character model

Figure 8 shows our word model where 3 or less transitions start from each state in order to consider the various configurations of the word segmentation. In this approach, the transitions do not express straightly the segmented characters but they are purposed to assimilate the over-segmentation and the interaction between characters in a word. Considering upper-case and lower-case month words, the lexicon contains 24 HMMs. A word is classified as upper-case if it contains at least half number of upper-case characters. The mixed case was not schemed in this study, because of the small size of the training database. For a given model, the number of states is determined during the training step.

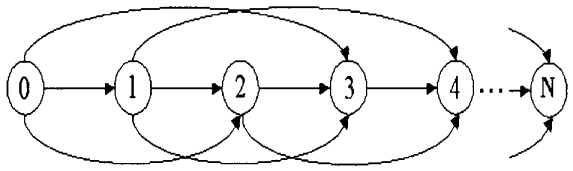

Figure 8: Word model with N states

4.2 Training

The goal of the training step is based on the best estimation of the models parameters for a given training set of observation sequences. These settings are provided by a variant of the *Baum-Welch* algorithm with the *cross-validation* approach [8]. Two sets are used: the models parameters are adjusted during the training step (with the training set) and the validation step computes the models performance in generalization (with the validation set).

For the analytical approach, each word model is built by the concatenation of the appropriated character models. In this case, the last state of a character model becomes the

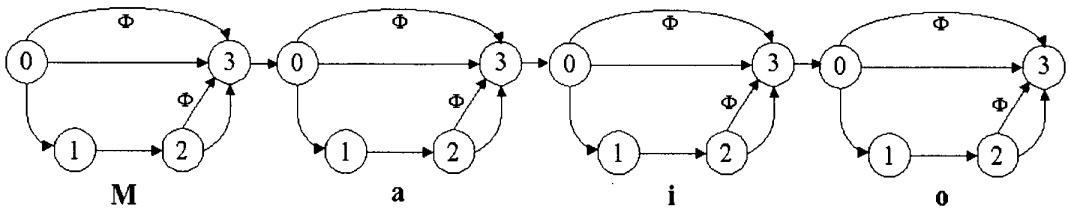

Figure 9: Training model of class "Maio" (May)

initial state of the next model, and so on, as shown in Figure 9.

4.3 Recognition

Our recognition approaches are based on the maximization of the *a posteriori probability* that a word w has generated an unknown observation sequence O, such as:

$$P(\overline{w}|O) = \max_{w} P(w|O)$$

Applying Bayes rule and after simplification, the recognition decision becomes equivalent to maximizing the joint probability:

$$P(\overline{w}|O) = P(O|w)P(w)$$

where $P(w)$ is the *a priori probability* of the word w (class distribution in the training set). In the current recognition module, we are not considering $P(w)$.

In the analytical approach, the word construction follows the same paradigm defined in the training step. However, as no information in recognition is available on the handwritten style (upper-case,lower-case), we adopted three word configurations: totally upper-case, totally lower-case and first character in upper-case and the others in lower-case. Thus, the estimation of a word probability is given by the probability combination (sum) of the three models, considering the *a priori probability* of each word model. In a same manner, the holistic models are also combined for each word class.

5 Experiments and analysis

From our laboratory database which contains 2,000 images, we used 1,188 images for training and 408 images for the validation. The distribution of lower-case words is about 80%, against 20% for the upper-case words.

The experiments were held on the test set with 404 word images. We used both Viterbi and Forward algorithms to proceed the recognition, but Table 2 details the best results obtained only with the forward approach. Note that we did not consider any rejection in this test. The third and fourth columns detail the recognition rate for each word class ("1" for "Janeiro", etc...), for holistic (H) and for analytical (A) approach respectively. The last column (C) gives the result of the combination of both approaches. The last line expresses the averaged rate for each system. Considering both recognition systems, we can observe that the best score is provided by the holistic approach, with an averaged recognition score of 81.43%. This can probably be explained by the fact that the holistic models are more flexible, since the number of states in this case is not a priori fixed, but optimally derived during the training process. Moreover, the holistic models can assimilate better the interaction between the characters in the same word, due to the strong irregularity of the character segmentation in unconstrained words.

Table 2: Recognition results with test set

Class	n^0 of Images	H (%)	A (%)	C (%)
1	39	82.05	92.31	94.87
2	32	78.13	78.13	81.25
3	36	69.44	72.22	69.44
4	39	87.18	89.74	87.18
5	38	78.95	81.58	78.95
6	30	80.00	83.33	80.00
7	33	78.79	60.61	84.85
8	28	89.29	82.14	89.29
9	31	87.10	80.65	87.10
10	30	86.67	93.33	93.33
11	34	91.18	79.41	88.24
12	34	70.59	64.71	70.59
Total	404	81.43	79.95	83.66

One peculiarity of this context concerns the likelihood of various words, such as:

- The terminations in "eiro" for "Janeiro" and "Fevereiro";
- The terminations in "embro" for "Setembro", "Novembro" and "Dezembro";

- Almost all characters between "Junho" and "Julho" and between "Maio" and "Março".

Depending on the discrimination abilities of the primitive extractor, all these similarities can seriously affect the performance of the recognition system. In Table 3, we can observe the main confusions between these classes, such as "Março" (class 3) and "Maio" (class 5).

Table 3: Confusion matrix

Class	1	2	3	4	5	6	7	8	9	10	11	12
1	37	0	1	0	0	1	0	0	0	0	0	0
2	2	26	0	0	1	0	0	0	0	1	0	2
3	0	0	25	1	8	0	0	0	2	0	0	0
4	0	0	0	34	3	0	1	1	0	0	0	0
5	0	0	1	3	30	0	1	3	0	0	0	0
6	0	0	0	0	0	24	6	0	0	0	0	0
7	0	0	1	0	3	1	28	0	0	0	0	0
8	0	0	2	0	1	0	0	25	0	0	0	0
9	0	0	0	1	0	0	0	0	27	2	0	1
10	0	0	0	1	0	0	0	0	1	28	0	0
11	1	0	0	0	0	0	0	0	1	1	30	1
12	0	0	0	0	0	0	0	2	6	1	1	24

Figure 10 shows some examples of correct classification and Figure 11 shows some recognition errors which correspond to the main problems of our system. These problems correspond to under-segmentation, high character distorsion, lack of training samples, etc.

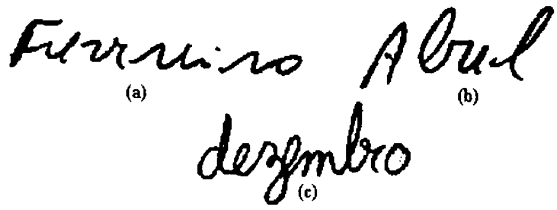

Figure 10: Examples of correct classification

Until now, we are strongly limited to compare our system with other applications dealing with the same context. Fan et al. [3] present one variant of the approach extended by Suen et al. in [7] which deal with dates on canadian checks written in french or in english. These works are more focused on field segmentation problems.

6 Conclusion and perspectives

This article describes the first stage of our recognition system applied to handwritten dates on brazilian checks. The

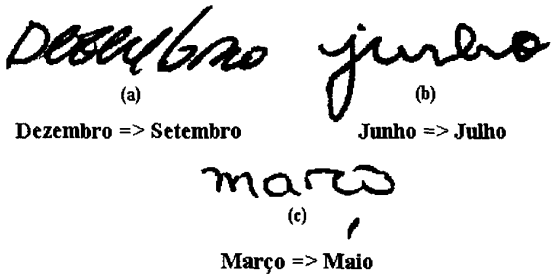

Figure 11: Examples of classification errors

challenge of this study remains in the problematic scheme of processing as a whole various data types such as words and digits in an omni-writer context.

Our first work we detail in this article deals with the recognition of the isolated month words, using HMMs with a limited lexicon. From the same primitive set, we use both holistic and analytical approaches to improve the global recognition rate of the system. Moreover, the analytical approach permits the creation of a character database with a greater number of training examples than in holistic approach. The segmentation module provides a grapheme sequence depending on the ligature localization in the cursive word.

The recognition combination provides an averaged rate of 83% with no rejection mode. We can consider these first results as satisfactory given the small size of our database and the limitations of the primitive extraction to discriminate upper-case characters.

Our future work is dedicated to the implementation of a new primitive set in order to improve the discrimination between the various writing styles. Moreover, we are modifying the analytical models to consider the mixture of upper-case and lower case characters in a word. Finally, we are studying a new approach to consider the date field as a whole in order to process the day, the month and the year in the same system.

References

[1] M. Y. Chen, A. Kundu, and S. N. Srihari. Variable duration hidden markov model and morphological segmentation for handwritten word recognition. *IEEE Transactions on Image Processing*, 4(12), December 1995.

[2] G. Dimauro, S. Impedovo, G. Pirlo, and A. Salzo. Automatic banckcheck processing: A new engineered system. *International Journal of Pattern Recognition and Artificial Intelligence*, pages 467–503, 1997.

[3] R. Fan, L. Lam, and C. Y. Suen. Processing of date information on cheques. In *Fifth International Workshop on Frontiers in Handwriting Recognition (IWFHR)*, pages 207–212, September 1996.

[4] A. M. Gillies. Cursive word recognition using hidden markov models. In *Fifth U.S. Postal Service Advanced Technology Conference*, pages 557–562, 1992.

[5] M. Gilloux, M. Leroux, and J. M. Bertille. Strategies for handwritten words recognition using hidden markov models. pages 299–304, 1993.

[6] M. A. Mohamed and P. Gader. Handwritten word recognition using segmentation-free hidden markov modeling and segmentation-based dynamic programming techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5), May 1996.

[7] C. Y. Suen, O. Xu, and L. Lam. Automatic recognition of handwritten data on cheques - fact or fiction ? *Pattern Recognition Letters*, 20(13):1287–1295, November 1999.

[8] A. El Yacoubi, R. Sabourin, M. Gilloux, and C. Y. Suen. Off-line handwritten word recognition using hidden markov models. In *Knowledge Techniques in Character Recognition*. CRC Press LLC, April 1999.