# High Performance Unconstrained Word Recognition System Combining HMMs and Markov Random Fields

George Saon and Abdel Belaïd

CRIN-CNRS

Bât Loria, Campus scientifique, B.P. 239

54506 Vandœuvre-Lès-Nancy Cedex, FRANCE

Phone: (33) 83 59 20 82

Fax: (33) 83 41 30 79

E-mail: {saon,abelaid}@loria.fr

**Abstract**

In this paper we present a system for the recognition of handwritten words on literal cheque amounts which advantageously combine HMMs and Markov random fields (MRFs). It operates at pixel level, in a holistic manner, on height normalized word images which are viewed as random field realizations. The HMM analyzes the image along the horizontal writing direction, in a specific state observation probability given by the column product of causal MRF-like pixel conditional probabilities. Aspects concerning definition, training and recognition via this type of model are developed throughout the paper. We report a 90.08% average word recognition rate on 2378 words and a 79.52% amount rate on 579 amounts of the SRTP[1] French postal cheque database (7031 words, 1779 amounts, different scriptors).

**Keywords:** Hidden Markov Models, Markov Random Fields, Handwritten Word Recognition

[1]Service de Recherche Technique de la Poste

# 1 Introduction

Nowadays, machine reading of bank cheques requires intensive study as it is an important commercial application. Given that a large number of cheques have to be treated each day in a bank, an automatic reading system would save much work even if it allows only about a half of them to be recognized with a high score. The tremendous recognition rate imposed by the French Post office for the non-rejected fraction of cheques leaves much space for further research in this field. The goal is to achieve a 0.01% error rate with less than 50% cheque rejection for manual processing. Concerning the literal amounts of cheques, even if the vocabulary size is reduced and the syntax is rigorous, word recognition remains difficult because of the totally unconstrained writings involved. A system for identification of literal amounts needs to operate at an omni-scriptor level since scriptor number is very high and their identities are unknown. For French postal cheques, several authors [16, 18, 20, 11, 10, 25, 26, 12, 6, 19, 9, 23, 4] have attempted to solve this problem and some of their results are summarized in [24] and in Table 3.

We will describe a new approach to handwritten word recognition (HWR) in a small lexicon, based on two dimensional Markov models. Since there is a certain similarity between HWR and speech recognition, techniques for recognition of unconstrained handwritten words can be borrowed from speech domain, which has been very active during the last decade. This is already the case with the hidden Markov models where their application benefits from the large experience achieved in speech recognition [22, 7, 3].

The use of hidden Markov models in HWR has led to interesting results for specific applications [3, 11, 19]. Nevertheless, HMM techniques imply sequential pattern processing prior to recognition by performing local observations commonly along the writing axis. This step generally contrasts with the 2D nature of writing (fundamental difference with speech) forcing researchers to enlarge HMM formalism. However, it was proved by Levin in [17] that a direct extension of the dynamic time warping algorithm (DTW), which is the basic mechanism of these models, to the plane, results in an NP-complete problem. By applying a class of constraints to the matching, the complexity can be pulled down to a polynomial one. A type of models issued from such a simplification are the PHMMs (planar- or pseudo HMMs) [2, 15].

The PHMMs are HMMs, where state observation probability is given by the emission probability of another HMM. They include a principal model composed of super-states

and secondary models associated to these super-states. For an image, the principal model will do the analysis according to one direction (for example, the vertical direction) and the secondary ones will do it along the other one. Several consecutive lines are thus associated to a given super-state assuming that they are tightly correlated and therefore analyzable possibly by the same secondary HMM. Even if these models are easy to implement, they are based on a line-independency hypothesis which does not always hold true in practice [2].

Some solutions in the literature consist of solving this independency problem by clustering homogeneous lines into classes using the *k-means* algorithm [15] or super-state equivalence classes [8]. Yet, this solution, being based on classification algorithms, does not solve fine dependency cases between consecutive lines.

We think that a perfectly two-dimensional model akin to our image recognition task would be more profitable. Therefore, we have studied the applicability of Markov random fields to HWR. Unlike PHMMs, these fields possess a real 2D structure as long as the probability of a random variable of the field is conditioned by the neighboring ones, and conditions at its turn probabilities on other sites.

Markov fields have been employed for a long time in statistical mechanics, the application of these models to images being more recent. They perform essentially low level tasks in image processing or artificial vision [13]. Until now, no attempt has been made to use MRFs in a complex recognition task such as HWR. This is due in our opinion to the fact that MRFs, as initially conceived, are only able to detect simple features in images like lines, edges of given orientations, textures, etc. This turns out to be insufficient for higher level recognition purposes. Our main idea was to provide MRFs with a "switching" mechanism between conditional probability distributions, in order to augment the capacity of the model to dynamically detect new features within the image (strokes of different orientations inherent to handwriting). This is done by tying column probability distributions to the states of an ordinary HMM; a transition to another state of the HMM implying an optimal change of these distributions in order to maximize word-image likelihood. One may make an analogy with the functioning of the visual striate cortex, where each neuron is sensitive to a particular stroke orientation.

We restricted our attention to the study of causal MRFs for two major reasons. First, as stated in [5], one cannot specify arbitrary conditioning neighborhoods for consistency reasons (existence of the joint field probability), whereas there are several theoretical

achievements on causal MRFs. On the other hand, recursive training and recognition procedures are more easily applicable on causal fields allowing a natural progression of the joint field mass probability calculus. The concept of causality may have different interpretations since the plane is not provided with a natural order.

Two types of causal MRFs are widely used in image processing: the Markov random mesh (MRM) [1] and the unilateral Markov random field also called non-symmetric half-plane Markov chain (NSHP) [21]. Jeng in [14] noted that NSHPs are more appropriate than MRMs when an accurate model for representing two dimensional data is required (MRMs are conditionally independent on $45^o$ diagonals which diminishes their capability to detect strokes having these orientations). With this in mind, we will focus on this type of model from now on.

The paper is organized as follows. In section 2, we define our proposed model and give the most important elements concerning training and recognition. Section 3 deals extensively with the experiments performed by describing the database in detail, the tests carried out and the results obtained. A discussion and concluding remarks are presented at the end of the paper.

## 2 Non-symmetric Half-plane Hidden Markov Models

Our main objective when conceiving the model was to avoid "hard" decisions taken by a high-level feature extraction step, which may sometimes alter the recognition process or, at least, render error recovery difficult. For example, in Figure 1.a, one may see that, because of noise and blur coming from image acquisition and binarization, the detection of the upstroke and the *t*-bar becomes almost impossible, hence word recognition very difficult. Conversely, parasite upstrokes and downstrokes may appear (see Figure 1.b) due to neighboring amount lines which will be detected and labeled by classical feature extraction algorithms, sometimes misleading the recognition system. By using a pixel level approach, only the hard decision is taken at the end for choosing, via Bayes rule, the most plausible word among the candidates. Of course, the disadvantage of this approach is the requiring of huge computational resources especially during the training phase, but it avoids the use of time-consuming preprocessing and feature extraction algorithms. We were obliged to reduce the dimensions of the word images by scaling them in order to

keep the resources within reasonable bounds. A pixel level approach is equally proposed in [6] by using a 3-layer neural network, trained by quick propagation to recognize cursive pieces of handwritten words.

We opt for a holistic approach (as in [12, 19]) since the vocabulary size of the target application is reduced (in French, 24 words are used to designate literal numbers plus the words "francs", "et" and "centimes"). Each input word may be thus viewed as an elementary pattern. No grapheme segmentation step is therefore required, avoiding problems commonly encountered of under- or over-segmentation. Moreover, word abbreviations frequently used ("frs" for "francs", "cts" or "ctimes" for "centimes", etc.) or misspelling ("cents" instead of "cent", "milles" instead of "mille", etc.), both resulting in letter deletions and/or insertions, are automatically taken into account within the same model if the global word shape is not affected too much. On the contrary, special word models for frequent abbreviations may be created.

## 2.1 NSHP Markov Random Fields

For the following definitions and properties, the reader may refer to [5]. We restrict our attention to random fields defined over a $m \times n$ integer lattice $L$. It is obvious that, in the context of HWR, $m$ and $n$ are the width and the height of the word image bounding box respectively. Each site $(i, j) \in L$ corresponds to a pixel. Let $X = \{X_{ij}\}_{(i,j)\in L}$ be a random field defined over the lattice $L$. $X^j$ stands for the column $j$ of $X$. Moreover, $P(X_{ij}|X_{kl})$ means implicitly the conditional probability of the realization $x_{ij}$ of $X_{ij}$ knowing realizations $x_{kl}$ of $X_{kl}$, that is $P(X_{ij} = x_{ij}|X_{kl} = x_{kl})$. Finally, the notation $P(X_{ij}|X_A)$, $A \subset L$, stands for $P(X_{ij}|X_{kl})$, $(k, l) \in A$.

Since we deal with binarized images, we only consider binary random fields, meaning that random variables take values of $\{0, 1\}$ (0-white pixel, 1-black pixel). According to the previous assumptions, a sample word image is naturally one possible realization of a random field.

Let us next define the NSHP Markov chain. Consider the following sets:

$$\begin{aligned} &\Sigma_{ij} = \{(k, l) \in L \mid l < j \text{ or } (l = j,\ k < i)\}, \\ &\Theta_{ij} \subset \Sigma_{ij} \end{aligned} \tag{1}$$

$\Sigma_{ij}$ is called the *non-symmetric half-plane* and $\Theta_{ij}$ the *support* of pixel $(i, j) \in L$. Both

types of sets are illustrated in Figure 2.

**Definition 1** *$X$ is a non-symmetric half-plane Markov chain if and only if:*

$$P(X_{ij}|X_{\Sigma_{ij}}) = P(X_{ij}|X_{\Theta_{ij}}), \qquad \forall\ (i,j) \in L \tag{2}$$

The joint field mass probability $P(X)$ may be computed following the chain decomposition rule of conditional probabilities:

$$P(X) = \prod_{j=1}^{n} P(X^j|X^{j-1}\dots X^1) = \prod_{j=1}^{n}\prod_{i=1}^{m} P(X_{ij}|X_{\Sigma_{ij}}) = \prod_{j=1}^{n}\prod_{i=1}^{m} P(X_{ij}|X_{\Theta_{ij}}) \tag{3}$$

Commonly, authors using NSHP Markov chains, choose for all $\Theta_{ij}$'s the same form, that is $\Theta = \{\Theta_{ij}\}_{1\le i\le m, 1\le j\le n}$, $\Theta_{ij} = \{(i-i_k, j-j_k) \mid 1 \le k \le P,\ j_k > 0 \text{ or } (j_k = 0, i_k > 0)\} \cap L$, where $P$ represents the number of neighboring pixels. The intersection with $L$ is due to boundary conditions. Note that definition of $\Theta_{ij}$'s satisfies (1). An example of a $\Theta_{ij}$ family is depicted in Figure 3. In section 3, we will study the influence of different $\Theta_{ij}$ collections on the average recognition score.

## 2.2 Definition of NSHP-like Hidden Markov Models

NSHP Markov chains can be implemented by HMMs if we consider the random field realization (word image) as an observation sequence of columns. In a specific state of the HMM, observation probability would be given by the column product of pixel conditional probabilities. A transition from a state of the model to another will result in changing the set of probability distributions, and in dynamically modifying feature sensitivity. After the training phase, the model will associate states to particular features (pixel distributions characterizing writing strokes) within the word image areas. For example, having specialized states for estimating the presence of upstrokes and downstrokes would be of great benefit for the recognition. The previously mentioned reasons, plus the fact that there are optimal training and recognition procedures, lead us to use HMMs to efficiently implement NSHPs. Figure 5 illustrates the implementation scheme of an NSHP model.

Let $\lambda$ be the HMM and let us rewrite equation (3) in terms of pattern likelihood with respect to $\lambda$:

$$P(X|\lambda) = \prod_{j=1}^{n} P(X^j|X^{j-1}\dots X^1,\lambda) = \prod_{j=1}^{n}\prod_{i=1}^{m} P(X_{ij}|X_{\Theta_{ij}},\lambda) \tag{4}$$

Suppose we have now a parallel stochastic state process associated to the columns $X^j$. We will denote this process by $Q = q_1 \dots q_n$, where the random variables $q_j$ take values in a finite set of states $S = \{s_1, \dots, s_N\}$. Equation (4) then becomes:

$$\begin{aligned} P(X|\lambda) &= \sum_{Q} P(X,Q|\lambda) = \sum_{Q} P(X|Q,\lambda)P(Q|\lambda) \\ &= \sum_{Q}\prod_{j=1}^{n} P(q_j|q_{j-1})P(X^j|X^{j-1}\dots X^1,q_j,\lambda) \\ &= \sum_{Q}\prod_{j=1}^{n} P(q_j|q_{j-1})\prod_{i=1}^{m} P(X_{ij}|X_{\Theta_{ij}},q_j,\lambda) \end{aligned} \tag{5}$$

under the assumption that $Q$ is a first order Markov process and that pixel distributions for column $j$ depend only on state $q_j$. Obviously, (5) bears a strong resemblance to the classical 1D HMM deduction (see, for example [22]) with the difference that we maintain 2D distributions, which we tie to specific states of our HMM. We are now able to explicitly define the notion of NSHP-HMM.

**Definition 2** *A non-symmetric half-plane hidden Markov model of order $P$ is defined by:*

- $\Theta = \{\Theta_{ij}\}_{1\le i\le m, 1\le j\le n}$, $\Theta_{ij} = \{(i-i_k, j-j_k) \mid 1 \le k \le P,\ j_k > 0$ *or* $(j_k = 0, i_k > 0) \cap L$, *where $P$ represents the number of neighboring pixels per site. $\Theta$ is called the* NSHP *support set collection (or simply the neighborhood set). All $\Theta_{ij}$'s are supposed to be ordered.*

- $V = \{0,1\}$, *the vocabulary (remember that we restrict ourselves to binary random fields). We denote a pixel realization of $X_{ij}$ by $x_{ij} \in V$.*

- $S = \{s_1, \dots, s_N\}$ *the set of the $N$ possible states of the model. We denote by $q_j \in S$ the state associated to column $X^j$.*

- $A = \{a_{kl}\}_{1\le k,l\le N}$, $a_{kl} = P(q_{j+1} = s_l | q_j = s_k)$, *the state transition probability matrix.*

- $B = \{b_{il}(x, x_1, \dots, x_P)\}_{1\le i\le m, 1\le l\le N}$, $x, x_1, \dots, x_P \in V$, *where* $b_{il}(x, x_1, \dots, x_P) = P(X_{ij} = x | X_{u_k v_k} = x_k,\ q_j = s_l)$, $(u_k, v_k) \in \Theta_{ij},\ 1 \le k \le P$ *the conditional pixel observation probabilities.*

- $\pi = \{\pi_i\}_{1\le i\le N}$ *where* $\pi_i = p(q_1 = s_i)$, *the initial state probabilities.*

For simplicity, we will denote henceforth an NSHP-HMM by $\lambda = (\Theta, A, B, \pi)$.

In the following, we show how to estimate the emission probability of a pattern (the image likelihood) and we give some elements concerning training and recognition.

## 2.3 Word Image Likelihood Calculus

An optimal evaluation of the likelihood $P(X|\lambda)$ is obtained using modified *forward-backward* functions. We will define the *forward* function $\alpha$ (*backward* function $\beta$ following a dual definition) as being the cumulated field probability until column $X^j$ of $X$ when ending in state $s_i$, $\alpha_j(i) = P(X^1 X^2 \dots X^j, q_j = s_i|\lambda)$;

1. $\alpha_1(i) = \pi_i \prod_{k=1}^{m} b_{ki}(X_{k1}, X_{u_1 v_1}, \dots, X_{u_P v_P}),\ (u_p, v_p) \in \Theta_{k1}, \qquad 1 \le i \le N$

2. $\alpha_j(i) = \left[\sum_{l=1}^{N} \alpha_{j-1}(l) a_{li}\right] \prod_{k=1}^{m} b_{ki}(X_{kj}, X_{u_1 v_1}, \dots, X_{u_P v_P}),\ (u_p, v_p) \in \Theta_{kj},$
   $1 \le i \le N,\ 2 \le j \le n$

3. $P(X|\lambda) = \sum_{i=1}^{N} \alpha_n(i)$

The complexity of the calculus of $\alpha$ is $\mathcal{O}(N^2 \times m \times n)$. We will see in section 3 that, by choosing a particular left-to-right architecture, it may decrease to $\mathcal{O}(N \times m \times n)$.

## 2.4 Multiple Sample Training

The goal is to determine the parameters $(A, B, \pi)$ of the model which maximize the product $\prod_{r=1}^{R} P(X^{(r)}|\lambda)$, where $X^{(r)}$ are sample word images used to train the model $\lambda$. Note that, as in the 1D case, there is no global optimization criterion and direct method. We use the maximum likelihood criterion (MLE) by performing Baum-Welch re-estimation.

Let us first define the *backward* function $\beta$:

1. $\beta_n(i) = 1, \qquad 1 \le i \le N,$

2. $\beta_j(i) = \sum_{l=1}^{N} a_{il} \prod_{k=1}^{m} b_{kl}(X_{kj}, X_{u_1 v_1}, \dots, X_{u_P v_P}),\ (u_p, v_p) \in \Theta_{kj},$
   $1 \le i \le N,\ j = n-1, \dots, 1$

- Since we opt for a left-to-right model architecture ($\pi_1 = 1$, $\pi_i = 0$, $2 \le i \le N$), initial state probabilities do not need to be re-estimated, that is: $\overline{\pi} = \pi$.

- For the state transition probability matrix $A$, we have:

$$\overline{a}_{il} = \frac{\sum_{r=1}^{R} \frac{1}{P_r} \sum_{j=1}^{n_r-1} \alpha_j^r(i) a_{il} \prod_{k=1}^{m} b_{kl}(X_{kj+1}^{(r)}, X_{u_1 v_1}^{(r)}, \ldots, X_{u_P v_P}^{(r)}) \beta_{j+1}^r(l)}{\sum_{r=1}^{R} \frac{1}{P_r} \sum_{j=1}^{n_r-1} \alpha_j^r(i) \beta_j^r(i)}, \qquad 1 \leq i, l \leq N \tag{6}$$

- For the conditional pixel observation probabilities:

$$\overline{b}_{il}(x, x_1, \ldots, x_P) = \begin{cases} \dfrac{\sum_{r=1}^{R} \frac{1}{P_r} \sum\limits_{\substack{j=1 \ s.t. \\ X_{ij}^{(r)} = x \ and \\ X_{u_k v_k}^{(r)} = x_k}}^{n_r} \alpha_j^r(l) \beta_j^r(l)}{\sum_{r=1}^{R} \frac{1}{P_r} \sum\limits_{\substack{j=1 \ s.t. \\ X_{u_k v_k}^{(r)} = x_k}}^{n_r} \alpha_j^r(l) \beta_j^r(l)}, & \text{if denominator} \neq 0 \\ b_{il}(x, x_1, \ldots, x_P), & \text{otherwise} \end{cases}$$

$$x, x_1, \ldots, x_P \in \{0, 1\}, \qquad 1 \leq i \leq m, \qquad 1 \leq l \leq N \tag{7}$$

where by $P_r = P(X^{(r)}|\lambda)$ we understand the emission probability of sample $X^{(r)}$ and by $n_r$ its length.

Let us take a closer look to equation (7). In fact, pixel probability re-estimation is done by performing an ML count of the number of times that a given pixel configuration is encountered. Having computed the functions $\alpha^r$ and $\beta^r$ for each sample image, the complexity of $B$ re-estimation is $\mathcal{O}(R \times \overline{n} \times N \times m \times 2^{P+1})$, $\overline{n}$ representing the average sample length. Note that all samples are supposed to have the same number of lines $m$ which necessitates a height normalization procedure prior to training or recognition. Another remark is that the complexity becomes exponential in the number $P$ of neighborhood pixels. In practice, it is this factor which affects the most the computational resources (memory and CPU time). With this in mind, we opt for small-order models with a large number of states.

### 2.5 Recognition

We chose a *model discriminant* approach by constructing an NSHP-HMM model for each word of the lexicon which will be trained only with samples of the given class. Recognition is performed simply by calculating the word image emission probability (image likelihood) for all models and by labeling the image according to the word model which produces the maximum probability (ML optimization criterion [22]). Bayes decision rule is employed to take into account model a priori probability since the number of word image samples may vary significantly from one model to another:

$$\lambda^* = \underset{\lambda \in \Lambda}{\operatorname{argmax}}\, P(\lambda|X) = \underset{\lambda \in \Lambda}{\operatorname{argmax}}\, \frac{P(X|\lambda)P(\lambda)}{P(X)} = \underset{\lambda \in \Lambda}{\operatorname{argmax}}\, P(X|\lambda)P(\lambda) \tag{8}$$

knowing that $P(X)$ is constant during recognition and therefore discardable. By $\Lambda$ we mean the set of word models and by $\lambda^*$ the model with maximum a posteriori probability.

## 3 Experiments and Results

Experiments were performed on the SRTP database which contains 1779 handwritten literal amount images (7031 words from 1779 different scriptors). Each image is provided with label file and horizontal word segmentation file. The last file was used for isolating the words within the amount image. The label file serves for direct comparison with the output of our recognition system (labels are supposed to be the exact transcription of the amount phrase).

### 3.1 Considerations on the data

Each amount image was scanned at 300 dpi from real postal cheques. The literal amount area was previously located and isolated. Horizontal and diagonal bars have been removed and images were binary thresholded. Examples of word images are shown at the end of the paper. In Table 1, we give some statistical information on the whole database concerning the number of samples for each word of the lexicon, the occurrence frequency and the number of samples per word used for training and testing. Amount word length varies between 2 and 11 with an average length of 4.

The only preprocessing that we apply is word-image height normalization. After this

step, all sample images will have the same number of lines $m$, but proportionally different widths.

## 3.2 Word Model Training

We randomly chose 4653 word images (1200 amounts, approximatively 2/3 of the database) for performing word model training. The precise number of samples used for each word class is given in Table 1. Next, we show how we chose the initial parameters for each model and the number of training steps required.

- *State number*: it is proportional to the average word length in pixel columns, $\overline{n}$, after height normalization. In practice, a number of states equal to $\overline{n}/2$ (varying from 11 for model "et" to 35 for "soixante" for $m = 20$ lines) gave the best recognition results.

- *State transitions*: we allow only transitions to the current or to the next state (strict left-to-right architecture). Initially, transition probabilities are equiprobable, that is $a_{ii} = a_{ii+1} = 0.5,\ 1 \leq i \leq N-1$.

- *Number of lines*: for computational trainability reasons, we limited this number to $m = 20$. Experiments were carried out with $m = 10$, $m = 15$ and $m = 20$ lines.

- *Model order (number of neighborhood pixels)*: we experimented models of order $P = 0 \ldots 4$ corresponding to the neighborhoods depicted in Figure 4.

- *Conditional pixel observation probabilities*:

$$b_{il}(x, x_1, \ldots, x_P) = \begin{cases} \dfrac{\sum_{r=1}^{R} \left| \left\{ j \mid 1 + \dfrac{(l-1)n_r}{N} \leq j \leq \dfrac{ln_r}{N},\ X_{ij}^{(r)} = x \text{ and } X_{u_k v_k}^{(r)} = x_k \right\} \right|}{\sum_{r=1}^{R} \left| \left\{ j \mid 1 + \dfrac{(l-1)n_r}{N} \leq j \leq \dfrac{ln_r}{N},\ X_{u_k v_k}^{(r)} = x_k \right\} \right|}, \\ \qquad\qquad \text{if denominator} \neq 0 \\ 0.5, \qquad\qquad \text{otherwise} \end{cases}$$

$$x, x_1, \ldots, x_P \in \{0, 1\}, \qquad 1 \leq i \leq m, \qquad 1 \leq l \leq N \tag{9}$$

All samples were divided in $N$ vertical bands of equal width. In (9), a normalized count of the number of pixel configurations $X_{ij}^{(r)} = x$ and $X_{u_k v_k}^{(r)} = x_k$, $(u_k, v_k) \in \Theta_{ij}$, within band

$l$ is performed over all samples $X^{(r)}$. The number of iterations (generally less than 10) varies from one model to another depending on the recognition rate obtained on training samples. The recognition rate threshold is determined empirically in function of the model order and the image height.

Figure 6 gives us visual feedback on the real learning capabilities of the word models. The grey levels code the probability of black pixels, and depend upon the state and the line index of the NSHP-HMM. The prototypes were obtained using models of order 3 trained with samples of height $m = 30$ (20 iteration steps). One may observe that, even if several hundreds of samples were used to generate a given prototype, the model is able to focus on pixel distributions characterizing specific writing strokes such as up- or downstrokes (for example "quatre", "neuf", "dix", "cinquante", "mille"), holes ("quarante", "soixante", "cent"), $t$-bars ("quarante", "soixante", "cent") or even $i$-dots ("six", "soixante").

## 3.3 Word and Amount Recognition

Recognition was done on 2378 words (579 amount images, representing roughly 1/3 of the database). The precise number of samples for each word is given in Table 1. We can see in Figure 8 the evolution of the average word recognition rate function of the image height and the order of the model. Generally, the score increases with the order of the models and with the height of the images (there is less data loss for a 20-line scaling). However, one exception appears in the case of the 0-order model (no neighborhoods at all), where normalizing images to 10 lines gives better results than 15 or 20 lines. The reason for this comes from the weak learning capacity of these models, which makes them more accurate on less data. As shown in Table 4, we finally obtain a top 1 average word recognition rate of 90.08% and a top 3 of 92.60%.

The time spent on a SPARC station 5 for recognizing the 2378 sample images (height of 20 lines) using a 4-order NSHP-HMM was 305.76 sec, that is an average of 128.58 msec per word image. Memory requirements are estimated at 145M (all word images scaled to 20 lines and the parameters of the 4-order NSHP-HMMs).

We added a phrase level to our word recognition system in order to see how well it performs on complete cheque amounts. It is based on stochastic grammars as described in [23]. Word segmentation limits within the amount phrase are supposed to be known. We report a 79.52% top 1 amount recognition accuracy (6.9% rejection rate) as shown in

Table 2. This score could be compared to the one obtained by Simon in [26] in case of amounts with unique segmentation (74.5% recognition in first position). In Figure 7, we show several images of amounts with their recognition results.

In Table 3, we give the word and amount recognition rates (where mentioned) for some existing systems in the literature. All the illustrated results were obtained on the same database. However, it is difficult to judge the different recognition rates since the partitioning in training and test sets differs from one system to another. A system is identified by its main author initials, the year and the reference to the article which describes it.

# 4 Conclusion

In this paper we have described a new approach to handwritten word recognition which combines causal MRF two dimensional modeling and HMMs. The word image is viewed as a random field realization which, at its turn, is considered to be an observation sequence of pixel columns. The emission probability of this sequence (image likelihood) is calculated using state dependent conditional pixel probabilities.

In opposition to PHMMs, a weakness of these models may be the requiring of height (or width) normalized input images. Nevertheless, the results obtained on a real database of unconstrained words are extremely satisfactory (90.08% first choice on 7031 words of 1779 scriptors). These results can be improved in the future by increasing the number of samples for training (which is highly insufficient for some less frequent words). Presently, we are working on the fine tuning of the models by applying corrective training for the misrecognized samples used during the training step. In parallel, we try to put into practice deleted interpolation techniques by tying together different sets of parameters (states of the HMM, pixel probabilities within states or within lines, etc.).

Future development will concern the automatic inference of neighborhoods in order to retain only informative pixels. One way to do so, is to use Akaike or Rissanen information criteria suited to two-dimensional data. Another problem which also needs to be addressed is the efficiency of our parameter estimation method MLE compared to methods based, for example, on maximum mutual information estimation.

# References


[1] K. Abend, T. J. Harley, and L. N. Kanal. Classification of Binary Random Patterns. *IEEE Trans. Inform. Theory*, 1(11):538–544, 1965.

[2] O. E. Agazzi and S. Kuo. Hidden Markov Model Based Optical Character Recognition in the Presence of Deterministic Transformation. *Pattern Recognition*, 26(12):1813–1826, February 1993.

[3] M. Y. Chen and A. Kundu. An Alternative to Variable Duration HMM in Handwritten Word Recognition. In *Proc. IWFHR-3*, pages 82–91, Paris, 1993.

[4] J. P. Crettez, M. Gilloux, and M. Leroux. What the Writer's "hand" Tells the Reader's "Eyes". In *Actes du* $4^{eme}$ *Colloque National sur l'Ecrit et le Document*, pages 291–296, Nantes, France, July 1996.

[5] H. Derin and P. A. Kelly. Discrete-Index Markov-Type Random Processes. *Proceedings of the IEEE*, 77(10):1485–1510, 1989.

[6] J. P. Dodel and R. Shinghal. Symbolic/Neural Recognition of Cursive Amounts on Bank Cheques. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 15–19, Montréal, 1995.

[7] A. M. Gillies. Cursive Word Recognition Using Hidden Markov Models. In *USPS'92*, pages 557–562, 1992.

[8] M. Gilloux. Reconnaissance de chiffres manuscrits par modèle de Markov pseudo-2D. In *Actes du* $3^{eme}$ *Colloque National sur l'Ecrit et le Document*, pages 11–17, 1994.

[9] M. Gilloux, B. Lemarié, and M. Leroux. A Hybrid Radial Basis Function Network/Hidden Markov Model Handwritten Word Recognition System. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 394–397, Montréal, 1995.

[10] M. Gilloux and M. Leroux. Recognition of Cursive Script Amounts on Postal Cheques. In *First European Conference dedicated to Postal Technologies*, pages 705–712, June 1993.

[11] M. Gilloux, M. Leroux, and J. M. Bertille. Strategies for Handwritten Words Recognition Using Hidden Markov Models. In *Second International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 299–304, Tsukuba, City Science Japan, 1993.

[12] D. Guillevic and C. Y. Suen. Cursive Script Recognition Applied to the Processing of Bank Cheques. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 11–15, Montréal, 1995.

[13] F. Heitz and P. Bouthemy. Multimodal Motion Estimation and Segmentation using Markov Random Fields. In *Proceedings of the 10th International Conference on Pattern Recognition (ICPR'90)*, volume 1, pages 378–383, Atlantic City, New Jersey, USA, 16-21 June 1990.

[14] F. C. Jeng and J. W. Woods. On the Relationship of the Markov Mesh to the NSHP Markov Chain. *Pattern Recognition Letters*, 5(4):273–279, 1987.

[15] S. Kuo and O. E. Agazzi. Keyword Spotting in Poorly Printed Documents Using Pseudo 2-D Hidden Markov Models. *IEEE Transactions on PAMI*, 16(8):842–848, 1994.

[16] M. Leroux, J. C. Salome, and J. Badard. Recognition of Cursive Script Words in a Small Lexicon. In *First International Conference on Document Analysis and Recognition (ICDAR'91)*, pages 774–782, Saint Malo, France, November 1991.

[17] E. Levin and R. Pieraccini. Dynamic Planar Warping for Optical Character Recognition. In *IEEE-ICASSP*, volume III, pages 149–152. IEEE, 1992.

[18] J. V. Moreau, B. Plessis, O. Bourgeois, and J. L. Plagnaud. A Postal Check Reading System. In *First International Conference on Document Analysis and Recognition (ICDAR'91)*, pages 758–766, Saint Malo, France, November 1991.

[19] C. Olivier, T. Paquet, M. Avila, and Y. Lecourtier. Recognition of Handwritten Words Using Stochastic Models. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 19–24, Montréal, 1995.

[20] T. Paquet and Y. Lecourtier. Handwriting Recognition: Application on Bank Cheques. In *First International Conference on Document Analysis and Recognition (ICDAR'91)*, pages 749–757, Saint Malo, France, November 1991.

[21] D. Preuss. Two-Dimensional Facsimile Source Coding Based on a Markov Model. *NTZ 28*, 5(4):358–363, 1975.

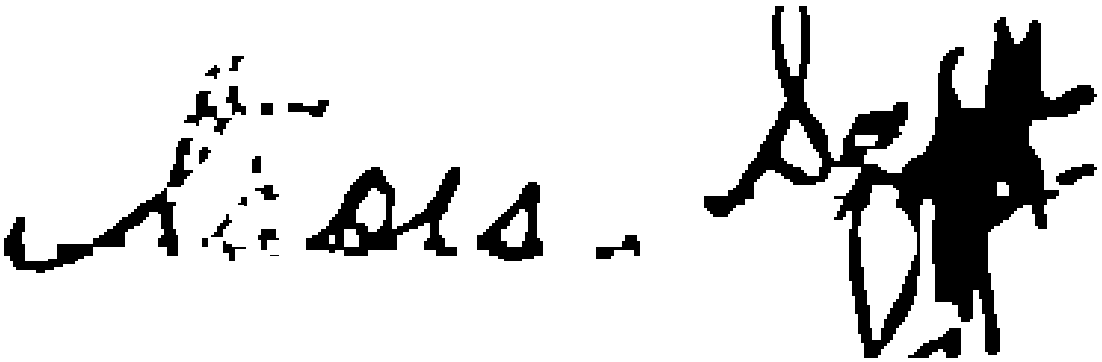
[22] L. R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), February 1989.

[23] G. Saon, A. Belaïd, and Y. Gong. Stochastic Trajectory Modeling for Recognition of Unconstrained Handwritten Words. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 508–511, 1995.

[24] A. W. Senior. Off-line handwriting recognition: a review and experiments. Technical report, Cambridge University Engineering Department, Cambridge, 1992.

[25] J. C. Simon. On the Robustness of Recognition of Degraded Line Images. In *First European Conference dedicated to Postal Technologies*, pages 695–696, June 1993.

[26] J. C. Simon, O. Baret, and N. Gorski. A System for the Recognition of Handwritten Literal amounts of checks. In *Internal Association for Pattern Recognition Workshop on Document Analysis System (DAS'94), Kaiserlautern, Germany*, pages 135–155, September 1994.



(a) (b)

Figure 1: Degraded word images.

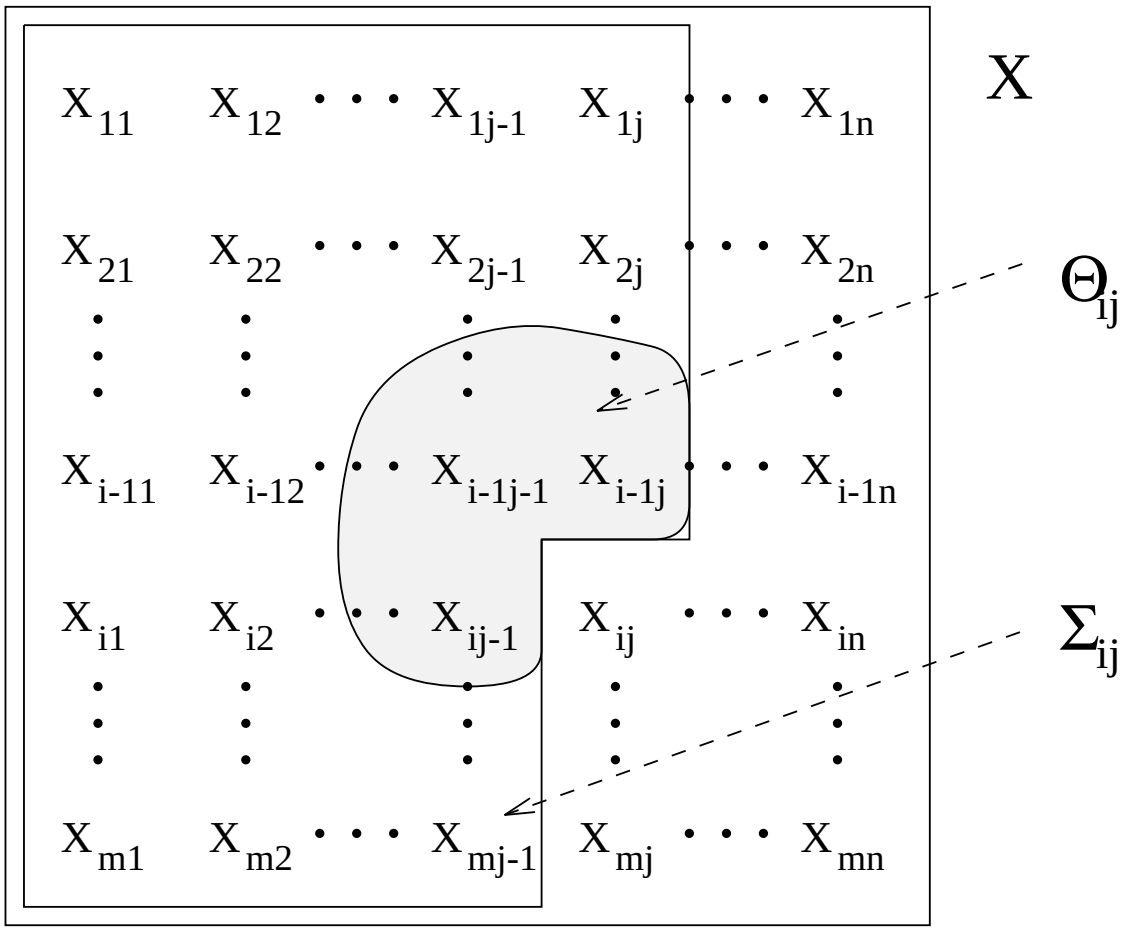



Figure 2: Sets of pixels related to site $(i, j)$.

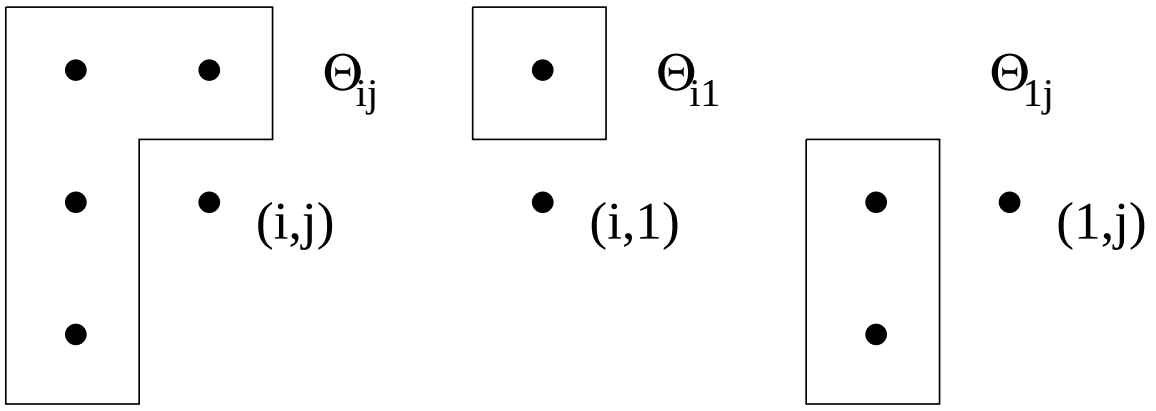



Figure 3: Example of a neighborhood family $\Theta_{ij} = \{(i-1,j),(i,j-1),(i-1,j-1),(i+1,j-1)\} \cap L$.

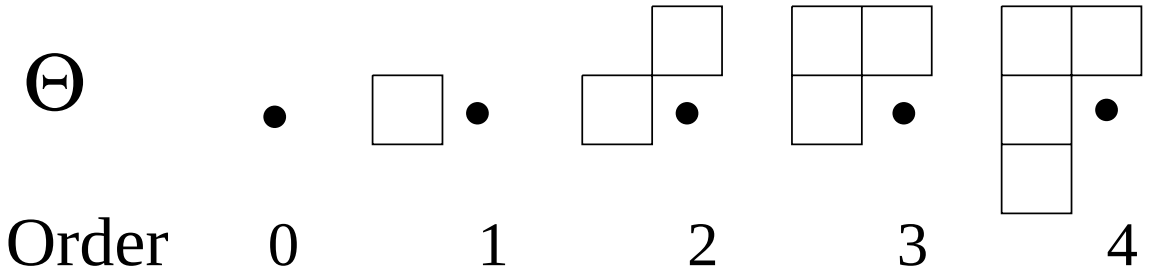



Figure 4: Various neighborhood patterns considered during testing.

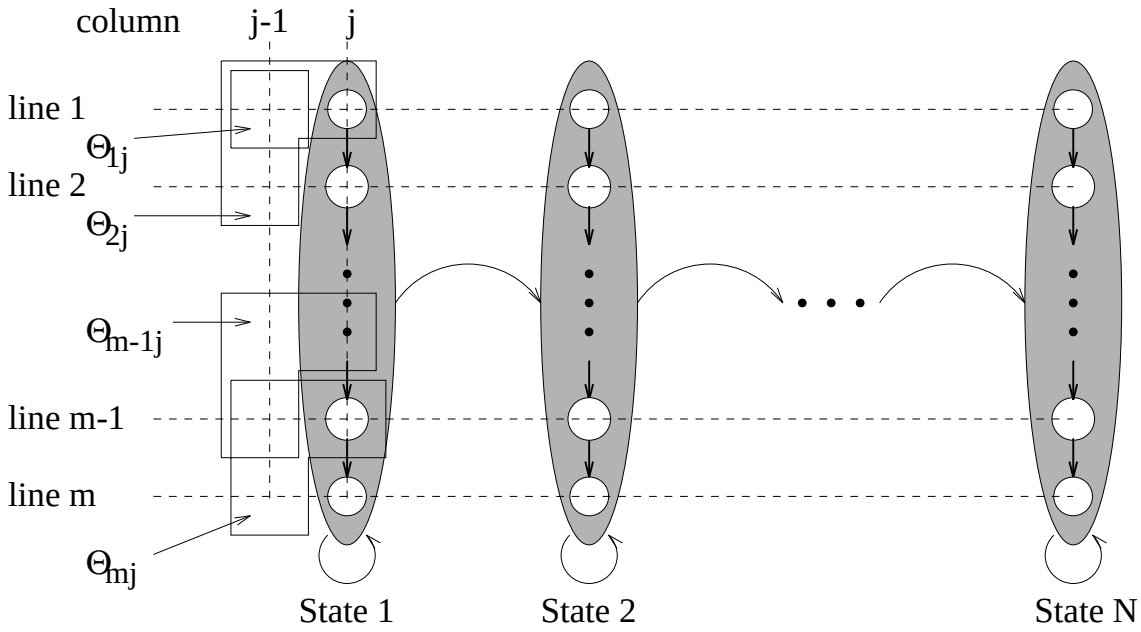



Figure 5: Architecture of an NSHP-HMM model.

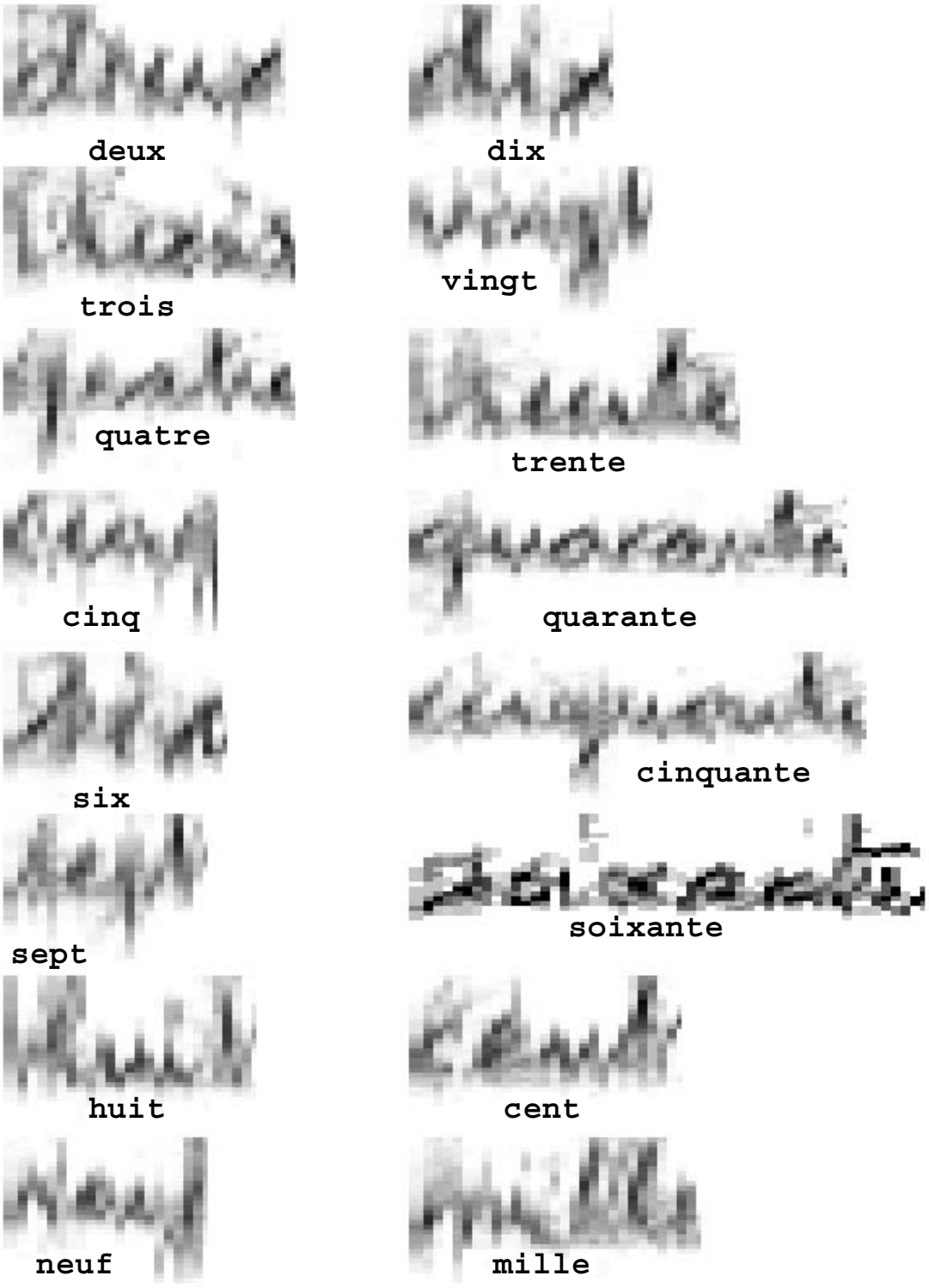
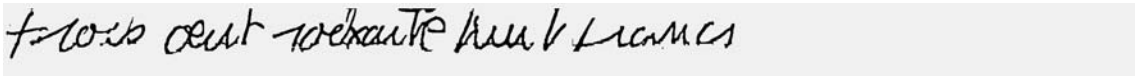
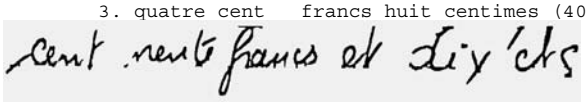



Figure 6: Word prototypes synthesis by NSHP-HMM models.



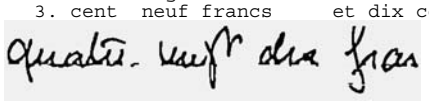
```
label:      trois cent soixante huit   francs (368.00F)

output: 1.  trois cent   francs huit centimes (300.08F)
        2.  trois cent soixante huit   francs (368.00F)
        3. quatre cent   francs huit centimes (400.08F)
```



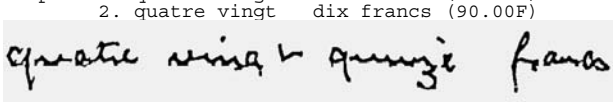
```
label:     cent  neuf francs     et dix centimes (109.10F)

output: 1. deux  cent  vingt francs dix centimes (220.10F)
        2. cent vingt francs     et dix centimes (120.10F)
        3. cent  neuf francs     et dix centimes (109.10F)
```



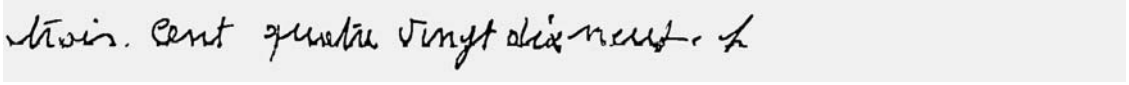
```
label:     quatre vingt   dix francs (90.00F)

output: 1. quatre vingt trois francs (83.00F)
        2. quatre vingt   dix francs (90.00F)
```



```
label:     quatre vingt quinze francs  (95.00F)

output: 1. quatre  cent quinze francs (415.00F)
        2. quatre  cent  vingt francs (420.00F)
        3. quatre vingt quinze francs  (95.00F)
```



```
label:     trois cent quatre  vingt dix neuf   francs (399.00F)

output: 1. trois cent quatre  vingt dix neuf   francs (399.00F)
        2. trois cent  vingt francs dix neuf centimes (320.19F)
        3. trois cent quatre francs dix neuf centimes (304.19F)
```

Figure 7: Examples of recognized amounts among the first 3 candidates.

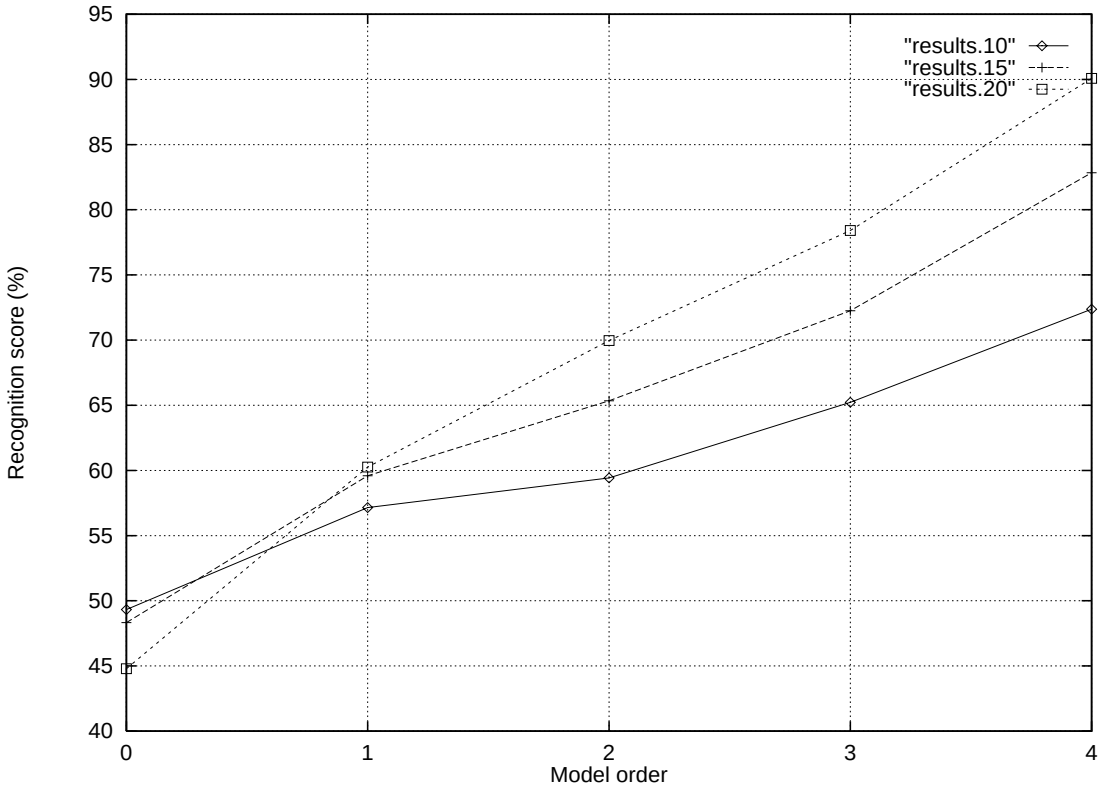



Figure 8: Evolution of the recognition score.

| Word | Nr. of samples | Occurence frequence | Nr. training | Nr. testing |
|---|---|---|---|---|
| UN | 28 | 0.41% | 20 | 8 |
| DEUX | 425 | 6.04% | 283 | 142 |
| TROIS | 238 | 3.39% | 154 | 84 |
| QUATRE | 519 | 7.37% | 333 | 186 |
| CINQ | 256 | 3.64% | 167 | 89 |
| SIX | 99 | 1.42% | 61 | 38 |
| SEPT | 104 | 1.49% | 64 | 40 |
| HUIT | 115 | 1.64% | 78 | 37 |
| NEUF | 128 | 1.83% | 86 | 42 |
| DIX | 239 | 3.40% | 150 | 89 |
| ONZE | 14 | 0.21% | 12 | 2 |
| DOUZE | 37 | 0.54% | 27 | 10 |
| TREIZE | 18 | 0.27% | 12 | 6 |
| QUATORZE | 21 | 0.31% | 16 | 5 |
| QUINZE | 63 | 0.91% | 38 | 25 |
| SEIZE | 16 | 0.24% | 8 | 8 |
| VINGT | 496 | 7.04% | 324 | 172 |
| TRENTE | 175 | 2.49% | 107 | 68 |
| QUARANTE | 126 | 1.80% | 96 | 30 |
| CINQUANTE | 154 | 2.20% | 97 | 57 |
| SOIXANTE | 232 | 3.30% | 152 | 80 |
| CENT | 1422 | 20.16% | 954 | 468 |
| MILLE | 230 | 3.27% | 160 | 70 |
| ET | 59 | 0.85% | 34 | 25 |
| FRANC | 1726 | 24.47% | 1176 | 550 |
| CENTIME | 91 | 1.30% | 44 | 47 |
| Total | 7031 | 100.00% | 4653 | 2378 |

Table 1: Statistical considerations on the database.

| Top | 1 | 2 | 3 | 4 | 5 | Reject |
|---|---|---|---|---|---|---|
| Training | 93.96% | 95.77% | 97.15% | 98.10% | 98.62% | 3.5% |
| Testing | 79.52% | 82.12% | 82.68% | 83.24% | 83.43% | 6.9% |

Table 2: Amount recognition accuracy on training and test sets.

| System | Word rec. rate | Phrase rec. rate |
|---|---|---|
| GIL93 [10] | 79% | 60% |
| GIL95 [9] | 83.7% | – |
| SIM94 [26] | 86.6% | 69.1% |
| OLI95 [19] | 72% | – |
| CRE96 [4] | 74.1% | – |
| SAO95 [23] | 82.83% | – |

Table 3: Recognition rates obtained by some existing systems.

| | height = 10 lines | | | height = 15 lines | | | height = 20 lines | | |
|---|---|---|---|---|---|---|---|---|---|
| Word | Top 1 | Top 3 | N1/N | Top 1 | Top 3 | N1/N | Top 1 | Top 3 | N1/N |
| UN | 100.00% | 100.00% | 8/8 | 100.00% | 100.00% | 8/8 | 100.00% | 100.00% | 8/8 |
| DEUX | 59.86% | 95.77% | 85/142 | 70.42% | 97.18% | 100/142 | 97.89% | 100.00% | 129/142 |
| TROIS | 60.71% | 85.71% | 51/84 | 76.19% | 89.29% | 64/84 | 91.67% | 98.81% | 77/84 |
| QUATRE | 72.04% | 88.71% | 134/186 | 87.63% | 93.01% | 163/186 | 87.10% | 95.16% | 162/186 |
| CINQ | 65.17% | 79.78% | 58/89 | 80.90% | 91.01% | 72/89 | 89.89% | 97.75% | 80/89 |
| SIX | 73.68% | 86.84% | 28/38 | 73.68% | 89.47% | 28/38 | 97.37% | 97.37% | 37/38 |
| SEPT | 45.00% | 70.00% | 18/40 | 87.50% | 100.00% | 35/40 | 100.00% | 100.00% | 40/40 |
| HUIT | 81.08% | 89.19% | 30/37 | 94.59% | 94.59% | 35/37 | 97.30% | 100.00% | 36/37 |
| NEUF | 42.86% | 69.05% | 18/42 | 73.81% | 85.71% | 31/42 | 88.10% | 97.62% | 37/42 |
| DIX | 73.03% | 83.15% | 65/89 | 86.52% | 92.13% | 77/89 | 96.63% | 97.75% | 86/89 |
| ONZE | 100.00% | 100.00% | 2/2 | 100.00% | 100.00% | 2/2 | 100.00% | 100.00% | 2/2 |
| DOUZE | 80.00% | 80.00% | 8/10 | 100.00% | 100.00% | 10/10 | 100.00% | 100.00% | 10/10 |
| TREIZE | 83.33% | 83.33% | 5/6 | 100.00% | 100.00% | 6/6 | 100.00% | 100.00% | 6/6 |
| QUATORZE | 80.00% | 80.00% | 4/5 | 80.00% | 100.00% | 4/5 | 100.00% | 100.00% | 5/5 |
| QUINZE | 56.00% | 64.00% | 14/25 | 96.00% | 96.00% | 24/25 | 88.00% | 88.00% | 22/25 |
| SEIZE | 87.50% | 87.50% | 7/8 | 100.00% | 100.00% | 8/8 | 100.00% | 100.00% | 8/8 |
| VINGT | 73.84% | 79.07% | 127/172 | 87.79% | 93.02% | 151/172 | 93.02% | 97.67% | 160/172 |
| TRENTE | 80.88% | 89.71% | 55/68 | 89.71% | 94.12% | 61/68 | 97.06% | 97.06% | 66/68 |
| QUARANTE | 60.00% | 63.33% | 18/30 | 63.33% | 80.00% | 19/30 | 83.33% | 86.67% | 25/30 |
| CINQUANTE | 66.67% | 70.18% | 38/57 | 85.96% | 89.47% | 49/57 | 80.70% | 80.70% | 46/57 |
| SOIXANTE | 81.25% | 87.50% | 65/80 | 85.00% | 88.75% | 68/80 | 68.75% | 68.75% | 55/80 |
| CENT | 84.83% | 87.82% | 397/468 | 87.61% | 89.32% | 410/468 | 92.74% | 93.59% | 434/468 |
| MILLE | 70.00% | 72.86% | 49/70 | 77.14% | 82.86% | 54/70 | 95.71% | 95.71% | 67/70 |
| FRANC | 68.55% | 68.55% | 377/550 | 78.36% | 78.36% | 431/550 | 88.55% | 88.55% | 487/550 |
| ET | 88.00% | 88.00% | 22/25 | 84.00% | 84.00% | 21/25 | 96.00% | 96.00% | 24/25 |
| CENTIME | 80.85% | 80.85% | 38/47 | 82.98% | 82.98% | 39/47 | 70.21% | 70.21% | 33/47 |
| Average | 72.37% | 80.57% | 1721/2378 | 82.84% | 88.06% | 1970/2378 | **90.08%** | **92.60%** | 2142/2378 |

Table 4: Word recognition scores for 4-order NSHP-HMM on testing set.