Segmentation-Driven Recognition Applied to Numerical Field Extraction from Handwritten Incoming Mail Documents

Clément Chatelain, Laurent Heutte, and Thierry Paquet

Laboratoire PSI, CNRS FRE 2645, Université de Rouen,
76800 Saint Etienne du Rouvray, France
clement.chatelain@univ-rouen.fr

Abstract. In this paper, we present a method for the automatic extraction of numerical fields (ZIP codes, phone numbers, etc.) from incoming mail documents. The approach is based on a segmentation-driven recognition that aims at locating isolated and touching digits among the textual information. A syntactical analysis is then performed on each line of text in order to filter the sequences that respect a particular syntax (number of digits, presence of separators) known by the system. We evaluate the performance of our system by means of the recall precision trade-off on a real incoming mail document database.

1 Introduction

Today, firms are faced with the problem of processing incoming mail documents: mail reception, envelope opening, document type recognition (form, invoice, letter, ...), mail object identification (address change, complaint, termination, ...), dispatching towards the competent service and finally mail processing. Whereas part of the overall process can be fully automated (envelope opening with specific equipment, mail scanning for easy dispatching, printed form automatic reading), a large amount of handwritten documents cannot be yet automatically processed. Indeed, no system is currently able to read automatically a whole page of cursive handwriting without any *a priori* knowledge. This is due to the extreme complexity of the task when dealing with free layout documents, unconstrained cursive handwriting, and unknown textual content of the document. Nevertheless, it is now possible to consider restricted applications of handwritten text processing which may correspond to a real industrial need. The extraction of numerical data (file number, customer reference, phone number, ZIP code in an address, ...) in a handwritten document whose content is expected (incoming mail document, see figure 1) is one particular example of such a realistic problem.

This paper presents a method for automatically extracting numerical fields from incoming mail documents in order to provide information about the sender: a phone number may be used to identify the customer, the ZIP code his location, the customer code is used to dispatch the document to the competent service,

H. Bunke and A.L. Spitz (Eds.): DAS 2006, LNCS 3872, pp. 564–575, 2006.

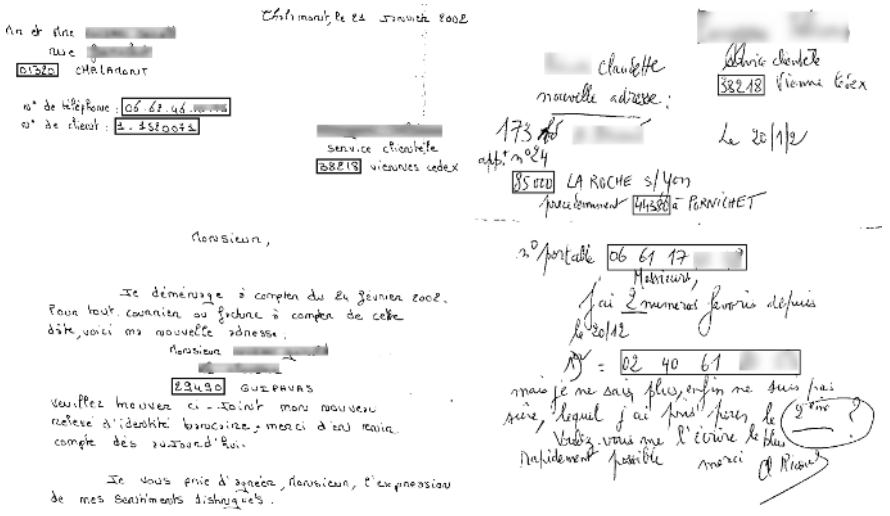

Fig. 1. Incoming mail document examples

etc. Our method is based on a syntactical analysis of the lines of text, to decide whether a numerical field is present or not. For that, a segmentation-driven recognition is performed on each component of the line, in order to locate the isolated and touching digits. The result of this recognition feeds a syntactical analyser that finds the best label sequence of each line of text, using the known syntax of the numerical field we want to detect (number of digits, presence of separator).

This paper is thus organized as follows. In section 2 we present an overview of the proposed system with a brief description of each processing stage. Section 3 presents the segmentation-driven recognition system. We present in section 4 our experimental results on a database of real handwritten incoming mail documents. Conclusion and future works are drawn in section 5.

2 Overview of the Proposed System

We aim at localizing and recognizing numerical sequences such as ZIP codes, custumer codes, phone numbers, etc. in unconstrained handwritten documents (see figure 1). It is a very challenging problem since we do not have any *a priori* knowledge about the documents that could help us to locate the fields: the numerical fields may be situated either in the header or in the body of the text. Hence, methods such as those used for example to locate a ZIP code in a handwritten adress [7, 9] do not suit our problem because they are based on strong *a priori* knowledge. Furthermore, numerical fields have no linguistic constraints: any digit can follow an other (see figure 2). Thus, our approach cannot be lexicon-directed as in many classical word recognition systems [8].

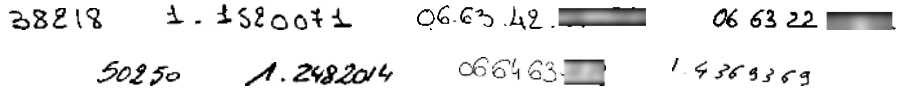

Fig. 2. Examples of numerical fields

The main idea of our approach is to exploit the known syntax of a numerical field to locate it in a text line. For example, a french phone number is always made of ten digits, with optional separators between each pair of digits. Thus, the extraction of a phone number in a text line consists in the detection of a sequence of ten digits with optional separators in the whole line sequence. This is performed by a syntactical analysis of the line sequence, which filters the syntactically correct sequences with respect to a particular syntax known by the system.

We have already presented in [16] and [15] a method for the automatic extraction of numerical fields from incoming mail documents based on this idea, but without recognition. In this paper, we propose to extend this approach by means of a segmentation-driven recognition, in order to extract and to recognize simultaneously the numerical fields from the documents.

Let us recall the principles of this approach (see figure 3). A line model is defined, which provides the syntactical constraints of a text line that may contain a numerical field. The model is composed of states describing the patterns that can occur in a text line: patterns that belong to a numerical field (digit, separator) or not (reject: word, fragment of word, etc.). Through a classification stage, we align the models on the text lines to filter the syntactically correct sequences. This implies that we need a classification system able to discriminate the components that belong to a numerical field (isolated and touching digits, separators) from the others (Reject).

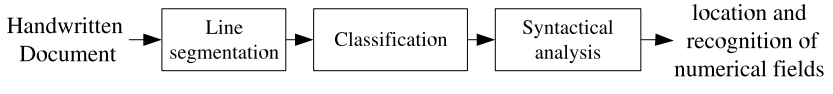

Fig. 3. Overview of the proposed system

The state definition and the classification system are the main differences between our previous approach and the one presented in this paper.

In the previous approach, we made the choice to avoid a segmentation task, and thus to directly classify the connected components with a restricted number of classes: “digit”, “double digit”, “separator” or “reject”, without trying to determine the numerical value of the digit and the double digit components. The numeral classification was performed in a second time, once the fields were extracted. Although this approach had the benefit to be fast, the classification task was difficult, principally due to the heterogeneity of the reject class, and in a lesser degree to the low inter-class variability between the digit and double

digit classes. Moreover, the system was unable to detect the fields that contain touching digits composed with more than two digits.

In the present approach, we propose to overcome these problems and to perform simultaneously the localisation and the recognition of the numerical sequences by applying a segmentation-driven recognition on each connected component. We process all the connected components of the document by considering them successively as an isolated, a double, or a triple digit, with a segmentation method and a strictly numeral classifier. Once the numeral recognition is performed, a reject class confidence value is estimated according to the digit classifier outputs to avoid the consideration of a reject class. The separator confidence value is estimated thanks to a small specific classifier. Hence the segmentation-driven recognition outputs a 3-level recognition hypothesis for each component, which are concatened to produce a trellis over all the line.

While performing a segmentation-driven recognition, the syntactical analysis has to find the best path in a 3-level trellis, whereas only one level was considered in the previous approach. The exploration of the trellis is performed by dynamic programming [2]. Figure 4 presents the models for a line which can contain a ZIP code, a customer code or a phone number, where arrows represent authorized transitions between states. Note that while in most cases a text line does not contain any numerical field, the model allows the line to be exclusively composed of reject patterns (i.e. does not include digits).

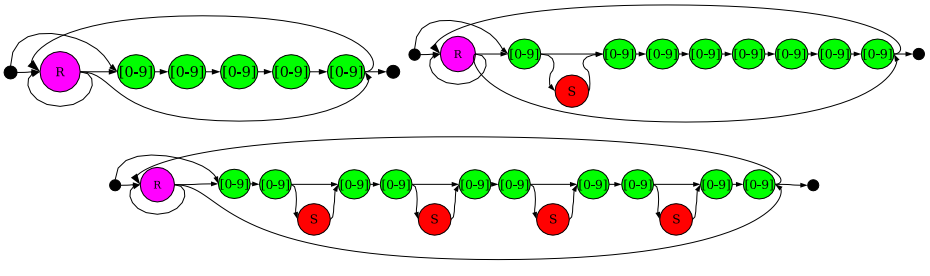

Fig. 4. Models for a line containing a ZIP code, a customer code and a phone number

Hence the three stages of our approach are the following ones (figure 3):

- **Line segmentation:** the connected components are extracted from the document and grouped into lines, according to a classical method [5]. The handwritten document is thus converted into sequences of connected components (see figure 5, and [16] for more details).
- **Segmentation-driven recognition:** during this stage, we search for numeral patterns. Numerical fields are mainly composed of isolated digits and separator components, but may also contain touching digits components (figure 2), which are hard to detect due to their high variability in size and shape. In this paper, we consider a segmentation-based strategy. Each connected component is submitted to a segmentation-driven recognition stage

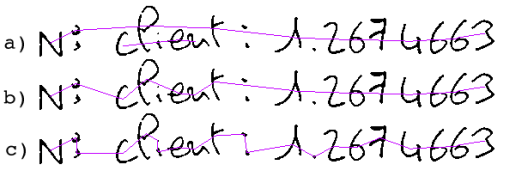

Fig. 5. The three steps for the line segmentation process: a) the big components are grouped together according to a distance criterion, b) alignments which are too close are merged, c) isolated components are grouped with the closest line

which recognize successively the component as an isolated digit, a double or a triple touching digit component. For that, we use a digit classifier with outlier rejection properties, able to output low confidence values when an outlier is encountered. The segmentation-driven recognition module thus outputs a trellis of recognition hypotheses. This stage is detailed in section 3.

– **Syntactical analysis:** this last stage filters the syntactically correct sequences with respect to a particular syntax known by the system, and searches among these sequences the best one according to the confidence values of the recognition hypotheses. Thus, a global decision is taken over the entire line and proposes a set of best paths which can contain or not a numerical field. The best path retrieval is achieved by the famous *forward algorithm* [2], a dynamic programming algorithm widely used within the framework of hidden Markov models.

3 Segmentation Driven Recognition

3.1 Principle

As mentionned in section 2, each connected component can either belong to a numerical field (digit, touching digit, separator) or not (word, fragment of word, noise, etc.). In this latter case, the precise nature of the component is unnecessary, and all these components should be considered as "reject". The classification problem is thus reduced to the discrimination of isolated digits, touching digits and separators from the remaining components (reject). We propose a two-stage recognition method for the component recognition.

First, a segmentation-driven recognition is performed in order to identify the numeral components: isolated or touching digit. Rather than considering a reject class in the classification problem, which is contraindicated due to the extrem heterogeneity of these patterns [12], we propose to design a classifier focused on the numerals. Hence, we avoid the difficult modelisation of a reject class. Thus, thanks to a 10-class digit classifier and a segmentation method, we investigate each connected component successively as being an isolated, a double or a triple digit (see figure 6). Note that touching digit may contain more than three digits, but they are extremely rare, and thus not considered in our work. Thus, the segmentation-driven recognition provides hypotheses of digit classes on 3 levels.

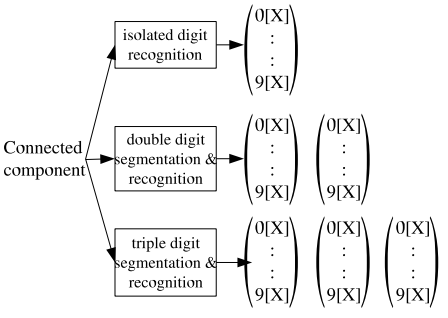

Fig. 6. Recognition of the components: Connected components are recognized as an isolated digit, a double or a triple digit. 'X' denotes a confidence value output by the digit classifier.

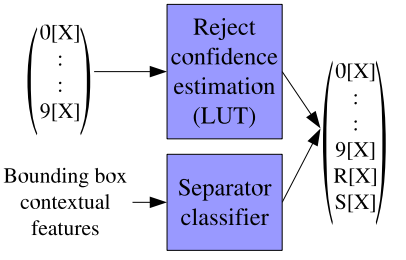

Fig. 7. In a second stage, reject and separator confidence values are estimated according respectively to the digit classifier outputs and a specific separator classifier

The second stage of the component recognition system is dedicated to the reject and separator identification among the digit recognition trellis. Indeed, as we do not consider reject and separator classes, the confidence values must be estimated during post processing identification stage. For that, we re-estimate the confidence values of each recognition hypothesis in the trellis by adding a reject and a separator confidence value (see figure 7). The reject confidence value is estimated with respect to the digit classifier confidence values, while the separator confidence value is estimated according to a specific classifier. One can see that we need a digit classifier whose confidence values are exploitable, i.e. these values must be high enough when an isolated digit is submitted, and low otherwise.

We now describe the components of the segmentation-driven recognition: the numeral classifier, the touching digit recognition method using a segmentation method, and the reject and separator confidence estimation procedure.

3.2 Handwritten Numeral Classifier

Usually, a good discrimination between the digit classes is the main criterion to design a handwritten digit classifier. This is due to the fact that the digit classifiers are mostly designed for the recognition of restricted and strictly numerical field (ZIP codes, amount on bankcheck, numerical field extracted from forms, etc.). In our case, the problem is rather different because the digit classifier is requested on each connected component of the whole page of handwriting. Thus, the digit classifier has to perform both:

- A discrimination task: when a digit is encountered, the classifier must be able to output the right digit class with a high confidence value.
- A detection task: as the digits to identify only represent a very small part of the documents, the classifier must have a strong outlier rejection ability in order to reject all the other connected components.

This second task is by far the most difficult: indeed, the very high variability of the outlier patterns (words, fragments of word, noise, stroke, touching digits, etc.) forbids the learning of a reject class for the classifier. Several techniques have been designed for the rejection of outlier: training a classifier with outlier data [12], modeling the target classes and perform a distance rejection strategy [17], one class classifiers [11], reject outliers with respect to the outputs of a classical digit classifier [14]. We have chosen this latter solution, applied on a MultiLayer Perceptron (MLP). This choice is motivated by the following reasons:

- As the classifier will have to process the segmentation hypothesis for all the connected components on the entire document, we cannot use a large time consuming classifier during the decision stage (this constraint prohibits for example multiclass SVM). An MLP well suits this condition because it is one of the fastest classifier during the decision stage.
- If the use of model-based classifiers (RBF, one class classifier, etc.) allows a distance-based rejection strategy, these classifiers provide generally quite poor results in discrimination, whereas MLPs perform very good results, especially in high dimensional spaces [3, 13].

We have thus designed a MLP trained on 130,000 digits, with a structural/statistical feature set. This feature set developped in our previous work [10], is made of 117 features and has been shown to achieve an efficient and robust discrimination of handwritten characters such as digits, uppercase letters or even graphemes.

In order to evaluate the capacity of our classifier to reject ouliers, we have built a database with both digits and outliers patterns. We consider the Receiver-Operating Characteristic (ROC) curve [6] which is a graphical representation

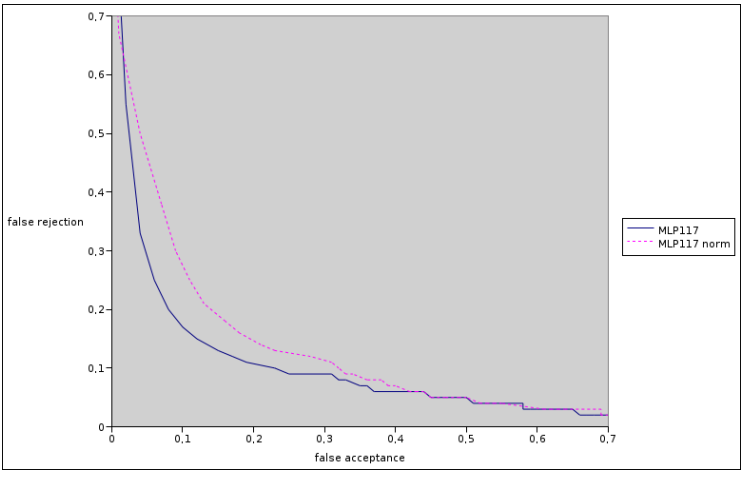

Fig. 8. ROC Curve for the MLP before and after the softmax function that scales MLP outputs into *a posteriori* probabilities

of the trade-off between the false negative (digit rejection) and false positive (outlier acceptation) rates for every possible cut off (confidence value of the first proposition of the MLP). Figure 8 shows the ROC curve for the MLP before and after the softmax function [1] that scales MLP outputs to *a posteriori* probabilities by means of a normalised exponential.

The resulting trade-off is slightly better before the softmax function, thus one can conclude that the analysis of the confidence values must be performed before the softmax function. On a test base of 33,000 digits, the classifier has a recognition rate of 97.78%, 99.20% and 99.60% (without rejection) respectively in TOP 1,2,3.

3.3 Touching Digit Recognition

The aim of this module is to find the best segmentation hypothesis when a double or a triple digit is submitted to the recognizer. For that, a descending segmentation-driven recognition is performed:

First, we make the hypothesis that we have to deal with a double digit connected component. In this case, the component is segmented in two parts according to a set of segmentation paths. These resulting digits are submitted to

Drop fall	ascending left	ascending right	descending left	descending right
cutting path				
digit classifier output	0[98] 8[82]	2[27] 8[35]	0[73] 8[36]	0[92] 8[34]
confidence product	81	09	26	32

Fig. 9. Double digit recognition: several cutting paths are generated and are submitted to a digit classifier. The path who maximizes the confidence product is retained (in this example, the first path).

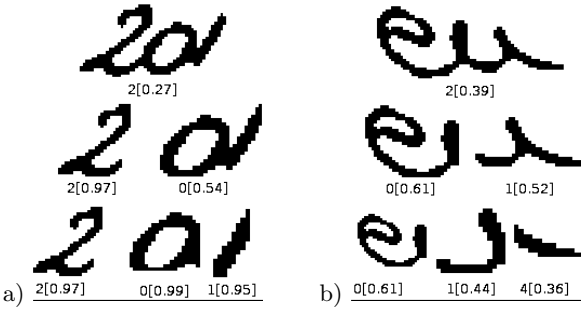

Fig. 10. Each component is recognized with a segmentation-driven recognition strategy as an isolated, double or triple digit. This provides a 3-level recognition trellis. Here, the recognition results are presented on a triple digits(a) and a fragment of word (reject)(b). The confidence values are lower for the reject components.

the digit recognizer, and the path which maximizes the product of confidence values is retained (see figure 9). Among these two digits, the one that has the lowest confidence value is regarded again as a double digit and is segmented in two parts according to the same principles of segmentation. Thus, we obtain for each component a 3-level recognition trellis (figure 10).

The segmentation paths are generated by the "drop fall" segmentation algorithm [4], which simulates the path of an acid drop falling from above the character and sliding along the contour. When the drop falls in a valley, it cuts the character and continues its path. This algorithm provides four possible paths, depending on the drop movement rules (left or right) and orientation (ascending or descending).

3.4 Reject and Separator Confidence Estimation

We need to estimate the probabilities for each pattern in the trellis to be a reject and a separator. The 10 digit classes recognition hypothesis are then converted in twelve classes: 10 numeral + Reject + Separator, each one associated to a confidence value (see figure 6). Once the two confidence values are estimated, the softmax function is applied on the twelve classes to approximate *a posteriori* probabilities [1].

The probabilities of reject and separator classes are estimated as follows:

Reject Confidence Value Estimation: the confidence value for the reject class is estimated with a Look Up Table (LUT) according to the confidence value of the first proposition of the digit recognizer. The LUT has been generated by considering the behaviour of the digit classifier on a database of 2300 digits and 4000 outliers. Statistics on the confidence value of the first proposition provides the LUT shown in figure 11.

Separator Confidence Value Estimation: the confidence value for the separator class is estimated by a small specific 2-class MLP classifier. This classifier

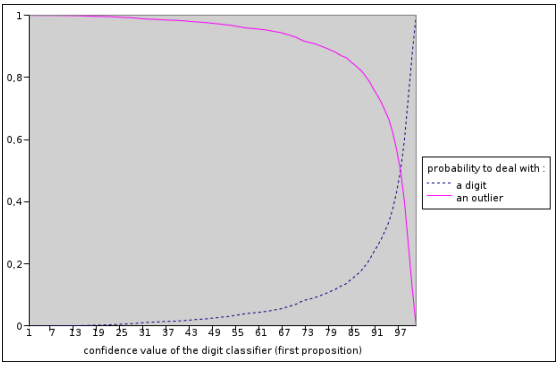

Fig. 11. Look up table for the reject confidence value. This LUT outputs the probability to deal with an outlier for a given digit confidence value.

have been trained on a database composed of separators and outliers (digits, word, fragment of word, noise), with a 9-contextual feature set described in [15]. The recognition rate is 96% (without rejection). As separators are always isolated and are usually small components, we assume that the second and third levels of the trellis cannot be a separator. Therefore, the confidence value of the separator class is directly used for the first level hypothesis of the trellis.

4 Results

The syntactical analysis is performed on each line of the documents, searching in the line trellis (see figure 12) the best path according to the syntactical constraints of the models defined on figure 4.

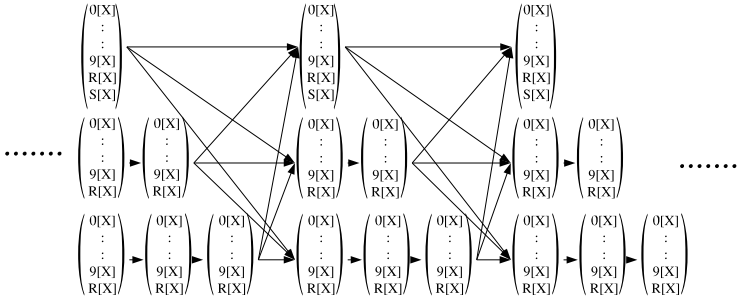

Fig. 12. Line trellis obtained by concatenation of the components trellis

We have evaluated our approach on a database of 293 handwritten incoming mail documents containing ZIP codes, phone numbers and customer codes. The syntactical analysis of the text lines is performed successively according to the three syntax models. A field is considered as "well recognized" if and only if all the components that belong to this field and only these ones have been labelled as the right numeral value of the field.

Let us recall that our method aims at extracting numerical fields for handwritten documents. Therefore, the best mean to analyse the performance of our system is the trade-off between recall and precision rates. The recall and precision rates are defined as:

recall = nb of fields well recognized / nb of fields to extract
precision = nb of fields well recognized / nb of fields proposed by the system

As the forward algorithm provides the n best alignment paths, a field well detected in "TOP n" means that the right recognition hypothesis for a field stands in the n best propositions of the syntactical analyser. It is obvious that the larger n, the more the recall increases, and the more the precision decreases. Table 1 shows the recall-precision trade-off for different values of n.

One can remark that our method is able to locate and recognize nearly 50% of the numerical fields from a document, with a precision of 20%. While increasing n,

Table 1. Recall and precision for the system when considering the n best paths

	TOP1	TOP2	TOP3	TOP4	TOP5
recall	0.49	0.53	0.56	0.59	0.60
precision	0.21	0.12	0.08	0.06	0.05

the system reaches a recall of 60% with a poor precision of 5%. Note that in an industrial application, the system could benefit from a customer database containing the researched fields and then would be able to filter the false alarms.

5 Conclusion and Future Works

In this article, we have presented a syntax directed method coupled with a segmentation-driven recognition applied to handwritten incoming mail documents, in order to locate and recognize numerical fields. It is a very challenging problem because we are faced with the classical problems encountered when dealing with totally unconstrained documents (lack of a priori information, high variability of handwriting). However, the system tested on a real handwritten incoming mail document database has given encouraging results as we obtain a recall of near 50%.

Our future work will focus on the improvement of the recall-precision tradeoff. In particular, we plan to improve the precision rate by designing a specific field verification method for filtering the false alarm, and to improve the recall rate by combining our previous approach (without segmentation [15]) with the one presented in this paper.

References

1. Bridle, J. S. " Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition ". In Neurocomputing: Algorithms, Architectures and Applications, F. F. Soulie and J. Herault, Eds., NATO ASI. 1990, pp. 227-236.
2. Rabiner, L. R. " A tutorial on hidden markov models and selected apllications in speech recognition ". In Readings in Speech Recognition. Kaufmann, 1990, pp. 267-296.
3. Bishop, C.M., Neural Networks for Pattern Recognition. Oxford University Press, 1995.
4. Congedo, G., G. Dimauro, S. Impedovo and G. Pirlo, " Segmentation of numeric strings ", ICDAR'95, vol. 2, 1995, pp. 1038-1041.
5. Likforman-Sulem, L. and C. Faure, "Une méthode de résolution des con its d'alignements pour la segmentation des documents manuscrits ", Traitement du signal, vol. 12, 1995, pp. 541-549.
6. Bradley, A.P., " The use of the area under the roc curve in the evaluation of machine learning algorithms ", Pattern Recognition, vol. 30, 1997, pp. 1145-1159.
7. Dzuba, Gregory, Alexander Filatov and Alexander Volgunin. " Handwritten zip code recognition. " . In ICDAR (1997), pp. 766-770.

8. Kim, G. and V. Govindaraju, " A lexicon driven approach to handwritten word recognition for real-time applications ", IEEE Trans. on PAMI, vol. 19, no. 4, 1997, pp. 366-378.
9. Srihari, S.N. and E.J. Keubert, " Integration of handwritten address interpretation technology into the united states postal service remote computer reader system ", ICDAR'97, 1997, pp. 892-896.
10. Heutte, L., T. Paquet, J.V. Moreau, Y. Lecourtier and C. Olivier, " A structural/ statistical feature based vector for handwritten character recognition ", Pattern Recognition Letters, vol. 19, 1998, pp. 629-641.
11. Tax, D.M.J. and Robert P. W. Duin. " Combining one-class classifiers". In MCS '01 (London, UK, 2001), Springer-Verlag, pp. 299-308.
12. Liu, C.L., K. Nakashima, H. Sako and H. Fujisawa, " Handwritten digit recognition using state-of-the-art techniques", IWFHR, 2002, pp. 320-325.
13. Liu, J. and P. Gader, " Neural networks with enhanced outlier rejection ability for off-line handwritten word recognition pattern recognition", Pattern Recognition, vol. 35, 2002, pp. 2061-2071.
14. Pitrelli, J.F. and M.P. Perrone, " Confidence-scoring post-processing for off-line handwritten-character recognition verification", ICDAR'03, vol. 1, 2003, pp. 278-282.
15. Chatelain, C., L. Heutte and T. Paquet, " A syntax-directed method for numerical field extraction using classifier combination ", 9th International Workshop on Frontiers in Handwriting Recognition, Tokyo, Japan, 2004, pp. 93-98.
16. Koch, G., L. Heutte and T. Paquet, " Numerical sequence extraction in handwritten incoming mail documents ", ICDAR, vol. 1, 2004, pp. 369-373.
17. Milgram, J., R. Sabourin and M. Cheriet, " An hybrid classification system which combines model-based and discriminative approaches ", 17th Conference on Pattern Recognition (ICPR2004), Cambridge, U.K., 2004, pp. 155-162.