A syntax-directed method for numerical field extraction using classifier combination.

C. Chatelain, L. Heutte and T. Paquet
Laboratoire PSI FRE CNRS 2645, Université de Rouen
76821 Mont-Saint-Aignan Cedex, France
{clement.chatelain, laurent.heutte, thierry.paquet}@univ-rouen.fr

Abstract

In this article, we propose a method for the automatic extraction of numerical fields in handwritten documents. The method exploits the syntax of a numerical field as an a priori knowledge to extract the connected component sequences from the document. For that, we have to label the connected components as "belonging to a numerical field" or not. We propose a method for discriminating the connected components, using different families of features and a combination of classifiers. A comparison between the results obtained with the combination of classifiers and our first approach [10] demonstrates the utility of combining different feature sets for discriminating classes with large intra-class variability.

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 yet be 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) is one particular example of such a realistic problem.

A numerical field is defined as a sequence of digits which often provides information about the sender: for example, a phone number may be used to identify the customer, the ZIP code his location, the customer code correctly dispatch the document to the competent service, etc. This paper proposes a method for their automatic extraction without recognition.

Indeed, a whole page recognition would be a very difficult and large time consuming task. The proposed method is an interesting alternative to the use of a digit recognizer prior to syntactical post processing. The proposed method will thus serve as a syntactical filter prior to recognition. Two components are required for this extraction task. The first one is dedicated to the labeling of the connected components. Labels are defined as Digit or Irrelevant handwritten information for the task. The second component is the syntactical analyzer that finds the best label sequence of each line of text using the known syntax of the numerical field we want to detect.

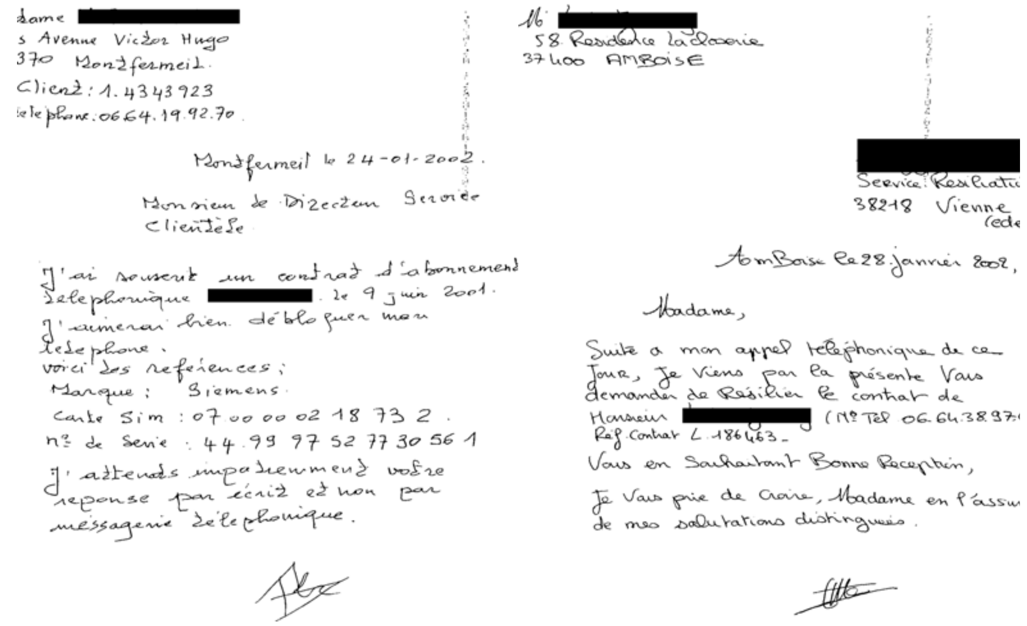
2. Overview of the proposed system

All the goal of this study is to design a system dedicated to the extraction of numerical data, such as zip codes, phone numbers or customer codes, in unconstrained handwritten documents. Figure 1 gives two examples of incoming mail documents. One can see that the fields of interest we are looking for can occur anywhere in the document (heading, body of text,...) or they can even sometimes be absent.

A naive approach would require that a digit recognizer processes the whole document. As we want to detect and recognize numerical fields, the recognizer should be able to recognize correctly the digits, whereas all the other connected components (letters, words, ...) should be rejected. Due to the presence of connected digits, a segmentation driven recognition would be required. However, multiple studies, especially dedicated to numerical amount recognition, have shown the difficulty of such an approach [6].

In order to avoid the full page recognition, we have rather turned towards an original approach that only label the connected components as "belonging to a numerical field" or not [10]. Once the component labeled, the syntactical structure of the expected numerical sequences

0-7695-2187-8/04 $20.00 © 2004 IEEE



dame [redacted]
s Avenue Victor Hugo
370 Montfermeil.
Client: 1.4343923
telephone: 0664.19.92.70.

Montfermeil le 24-01-2002.

Monsieur le Directeur Service
Clientèle

J'ai souscrit un contrat d'abonnement
telephonique [redacted] . le 9 juin 2001.
J'aimerai bien débloquer mon
telephone.
voici les references;
Marque : Siemens.
Carte Sim : 07 00 00 02 18 73 2 .
n° de Serie : 44 99 97 52 77 30 56 1
J'attends impatiemment votre
reponse par écrit et non par
messagerie telephonique.

M [redacted]
58 Residence La Closerie
37400 AMBOISE

[redacted]
Service Resiliatio
38218 Vienne
Cede

AmBoise le 28 janvier 2002,

Madame,

Suite a mon appel telephonique de ce
Jour, Je Viens par la présente Vous
demander de Résilier le contrat de
Monsieur [redacted] (N° Tel 06.64.38.97.
Ref. Contrat L.186463_
Vous en Souhaitant Bonne Reception,

Je Vous prie de Croire, Madame en l'assu
de mes salutations distinguées.

Figure 1: Incoming mail documents

is used to filter the fields of interest in the document. We can compare our method with the "lexicon directed" handwritten word recognition methods [8]. In these methods, the observation sequences are aligned on lexicon words, whereas in our method, the observation sequences are aligned on a particular syntax. Our method can thus be qualified as "syntax directed".

Our system is thus divided into the three following stages, once all the connected components have been extracted from the document: the first one is the extraction of lines of a text, based on a classical method [13] and therefore beyond the scope of this paper. The two remaining stages, i.e. component labeling and field extraction by syntactical analysis, are recalled in this section.

Connected component labeling: We are interested here in assigning to each connected component its unknown label. From a syntactical point of view, a numerical field is namely composed of digits and separators (point or dash). It can also contain touching digits which must be identified. As we do not want to perform recognition yet, it seems to be very difficult to perform segmentation. Thus, touching digits are considered as a class in our discrimination problem. All the other connected components must be considered as reject. Taking into account the above observations, four classes of connected components have to be considered for labeling: Digit (class "D"); touching digits ("DD"); separators ("S") and reject ("R"). We insist on the fact that these four classes are very difficult to discriminate due to the intra-class variability of the classes "D", "DD" and especially "R".

Field extraction by syntactical analysis: This last stage is crucial for the system as it will allow to verify that some sequences of connected components can be kept as candidates. Indeed, the numerical sequences we search for all respect one precise syntax (five digits for a French zip code, ten digits for a French phone number,...). The syntactical analyzer will therefore be used as a precise numerical field localizer able to keep the only syntactically correct sequences and reject the others. One can thus postulate that our numerical extraction method is "syntax directed". The localization of numerical fields within a text line is therefore achieved through the Viterbi algorithm [5], a commonly used algorithm for sequence alignment. To apply this algorithm, it is necessary to define a hidden Markov model. In our case the alphabet is reduced (only four classes, i.e. digit, double digit, separator and reject) and the numerical fields we search for are constrained by a strong syntax (zip codes, phone numbers and customer codes). We have thus chosen to define one model for each syntax, i.e. for each type of numerical field to extract within a line. The other models are built in the same way.

A French zip code is constituted of five digits, each one corresponding to a given state: D0, D1, D2, D3, D4. As a line of text may contain, in addition to the zip code field, words that must be in our case rejected, it is necessary to introduce an additional rejection state, denoted by R. A transition matrix is then built to reflect the probabilities of transition of one state towards the others. For example, if one is in D0 state, the only possible transition for a zip code is D0 towards D1, all others being forbidden. While arguing in the same way on all states, we get the following syntax model (figure 2):

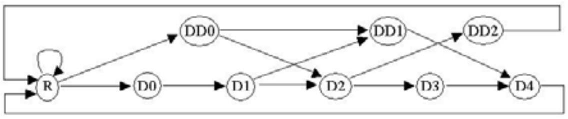


Figure 2: Syntax model for a zip code

Note also that to complete the model, we have moreover to define a matrix of initial and final states. This model is described in detail in [10].

A first application using a 9 contextual feature set and a K-NN classifier: To discriminate these four classes, a first approach described in [10] has been proposed, based on a set of contextual features, extracted from the bounding box of the connected components.

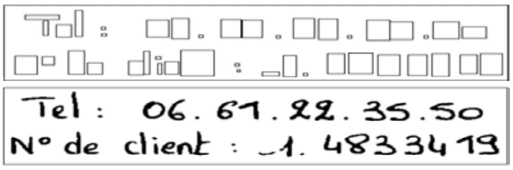


Figure 3: Connected component's bounding box for a phone number and a customer code

This approach is based on the assumption that numerical sequences are made of components whose size, spacing and position are quite regular. We can observe this phenomenon on fig. 3: the bounding boxes of the connected components which belong to numerical sequences are quite easy to localize provided we have an idea of the numerical sequence.

Let C be the connected component under investigation, C-1 and C+1 its left and right neighbors respectively; let Hc, Wc , Gcx and Gcy be respectively its height, width, the X and Y coordinate of its center of gravity. By taking into account height and width of C-1 and C+1 and related distances of C-1 and C+1 from C, the regularity/irregularity in height, width and spacing in the neighborhood of C can be measured through the following features:

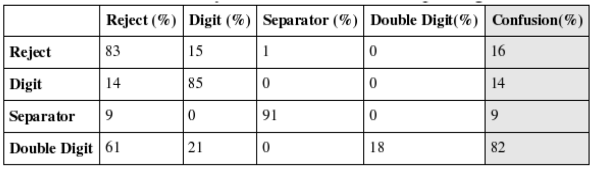
$$f_1 = \frac{H_{C-1}}{H_C} \quad f_2 = \frac{H_{C+1}}{H_C} \quad f_3 = \frac{W_{C-1}}{W_C} \quad f_4 = \frac{W_{C+1}}{W_C} \quad f_5 = \frac{H_C}{W_C}$$

$$f_6 = \frac{G_{Cx} - G_{Cx-1}}{W_C} \quad f_8 = \frac{G_{Cy} - G_{Cy-1}}{W_C} \quad f_9 = \frac{G_{Cy} - G_{Cy+1}}{W_C}$$

Once this 9-feature vector extracted, likelihood of each of the four classes can then be estimated using a K-Nearest Neighbor classifier. Its likelihood feeds a syntactical analyzer which keeps the only syntactically correct sequences. Let us call KNN9 this first classification method.

KNN9 limitations: A first limitation of the proposed method appears while processing a regular and script writing document. Indeed, our feature set is extracted on the bounding box of the connected component, because we have postulated that numerical fields have generally regular spacing and size. However, in the particular case of a regular and script handwritten document, this property is even true for the whole document (figure 4). According to this set of features based on the bounding box, it is impossible to distinguish numerical fields from text.

Let us now look at the KNN9 behavior. Table 1 shows the KNN9 confusion matrix. One can remark the strong confusions between reject and digit, separator and double digit classes. Moreover, the double digits are dramatically confused with Reject. We can explain this confusion by a simple example: let us consider a double and a simple digit which both belong to a numerical field, owning approximatively the same bounding box (fig. 5: "06" and "1"); in this particular case and according to our contextual features, there is no way to discriminate a simple digit from a double one without segmentation.



	Reject (%)	Digit (%)	Separator (%)	Double Digit(%)	Confusion(%)
Reject	83	15	1	0	16
Digit	14	85	0	0	14
Separator	9	0	91	0	9
Double Digit	61	21	0	18	82

Table 1: TOP1 confusion matrix for KNN9

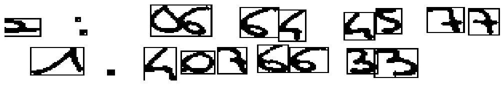


Figure 5: Double digit "06" and digit "

All this converges to show that the 9-feature set based on the bounding box is not sufficient enough to correctly discriminate the four classes involved.

3. Improvements on the connected component labeling

This section deals with improvements concerning the connected component labeling. First, two feature sets are extracted from the connected component, in addition to the contextual features. Then, connected component labeling is performed using a parallel combination of multilayer perceptron classifiers.

3.1 Feature set extraction based on intrinsic properties

It has been shown in the previous section that a feature set extracted from the bounding box could not achieve a good discrimination, especially in case of regular and script writing. Thus, we propose to extract a feature set that describe the intrinsic properties of the connected components to avoid this phenomenon.

Among the large number of feature extraction methods available in the literature [19], a commonly and successfully used feature set is the chaincode feature set extracted from the contours of the connected component

[9]. In addition to this feature set, we have also decided to use the statistical/structural feature set developed in our previous work [7]. This feature set is made of 117 features from 6 families and has been shown to achieve an efficient and robust discrimination of handwritten characters such as digits, uppercase letters or even graphemes [7].

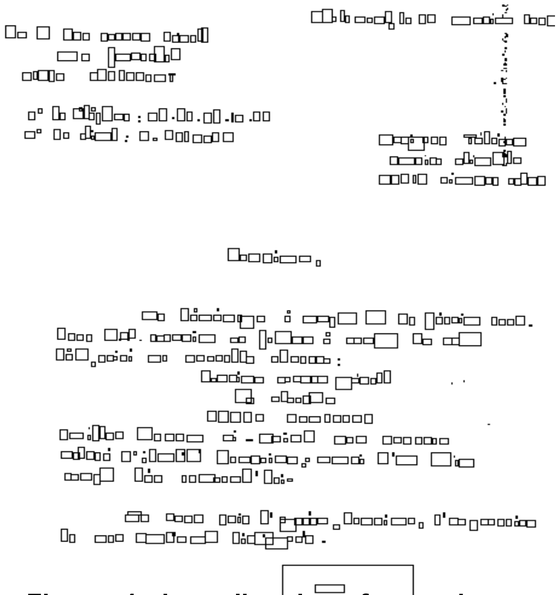


Figure 4: bounding box for an image of regular writing type

We now have at our disposal three feature sets to describe our connected components. A multilayer perceptron is then used to classify the connected components through each of these three feature sets.

3.2 Design of a Multilayer Perceptron (MLP)

MLP [1] is a statistical neural classifier widely used in handwritten character recognition [12]. As the MLP does not require large time consuming during the decision stage, this allows us to have a quite large number of features to describe the connected components. An other advantage is that the outputs of a MLP are estimates of Bayesian *a posteriori* probabilities [18]. This is an interesting property to feed the syntactical analyzer and to process sequences using the Viterbi algorithm.

We have thus designed one MLP classifier for each of the feature sets available. Let "MLP9" be the MLP trained on the 9 contextual/morphological feature set; "MLP128" the MLP trained on the chaincode feature set and "MLP117" the MLP trained on the 117 statistical/structural feature set. They are all designed on the same scheme:

- An input layer containing as many neurons as number of features : 9, resp. 128 and 117 neurons.
- One unique hidden layer, made of 20 (resp. 200 and 200) neurons.
- An output layer, containing as many neurons as classes in our discrimination task, i.e. 4 for all three feature sets.
- The activation function of each neuron is a sigmoid.

These multi-layer feed-forward neural networks are trained with the iterative back-propagation algorithm. Our MLPs have been trained on 13000 connected components from which the three feature sets have been extracted. This training set contains 7008 rejects, 4968 digits, 522 separators and 334 double digits. The test set is made up of 6000 connected components: 3609 "R", 2559 "D", 268 "S", 171 "DD". The confusion matrices of MLP9, MLP128 and MLP117 on the test set are respectively presented in tables 2, 3 and 4.

MLP9 (%)	Reject	Digit	Separator	Double Digit	Confusion (%)
Reject	74	25	1	0	26
Digit	20	80	0	0	20
Separator	16	0	84	0	16
Double Digit	75	24	0	1	99

Table 2: TOP1 confusion matrix for MLP9

MLP128 (%)	Reject	Digit	Separator	Double Digit	Confusion(%)
Reject	70	21	5	3	29
Digit	10	83	0	8	18
Separator	25	7	67	1	33
Double Digit	20	41	0	39	61

Table 3: TOP1 confusion matrix for MLP128

MLP117 (%)	Reject	Digit	Separator	Double Digit	Confusion(%)
Reject	82	12	4	3	21
Digit	15	78	0	7	22
Separator	34	1	65	0	35
Double Digit	21	37	0	42	58

Table 4: TOP 1 confusion matrix for MLP117

As we can see from tables 2, 3 and 4, each classifier has its own behavior. We can mention the high capacity of the MLP9 to detect the separators, whereas the MLP128 has the lowest confusion rate on Digit, as the MLP117 on the reject and double digit. These classifiers seem to be complementary to discriminate the four classes, and thus we aim at combining their output to give the best connected component labeling.

Once the three MLP trained, we try to choose the best combination of classifiers to discriminate the 4 classes.

3.3 Combination of classifiers

We now dispose of three MLP classifiers to discriminate the four classes Digit-Separator-Double Digit-Reject. As we can see in the previous section, none of the three classifiers can be considered as the best and only solution, but they are all of interest. It has been shown in previous studies [4, 21] that a combination of classifiers could improve recognition reliability by taking into account the complementarities between classifiers.

There are many way to combine classifiers, depending on the amount of information to combine [20]: *abstract-level* combination methods use the top candidate of each classifier ; *Ranked-level* combination methods use the entire ranked list of candidates and *Measurement-level* combination methods use the confidence value of each candidate in the ranked list. This last combination method also provides a confidence value, which is the information we need for processing sequences using the Viterbi algorithm. Thus, the outputs of our three MLPs are combined with a measurement method.

Different combination rules can be used to provide the final output [20]: maximum, minimum, median, product, linear combination are the most commonly used. Product (Prod) and mean (Mean) have been used in our tests. Combinations of these three classifiers have been benchmarked, we present in the next section the results on numerical fields detection on handwritten unconstrained documents.

4. Experimental results

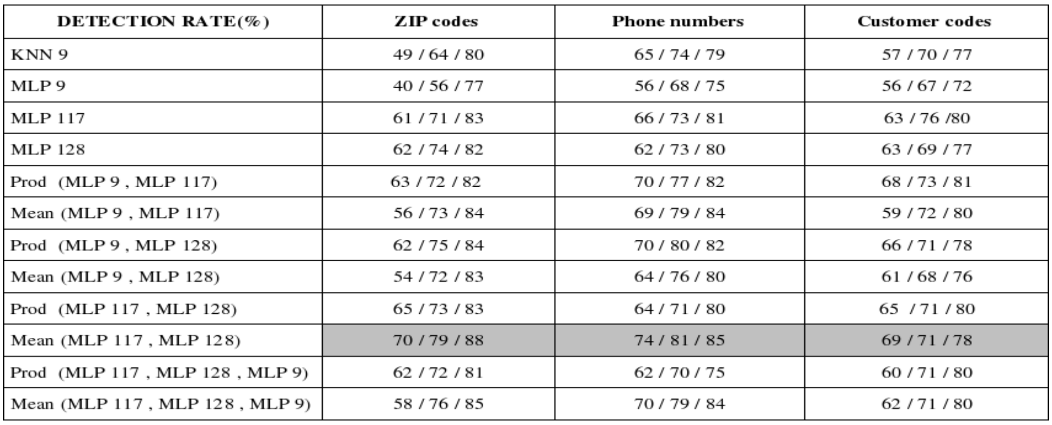
To determine which is the best combination for our specific problem, all the combinations have been benchmarked. Thus, we compare the results of the initial KNN9 with: 3 simple classifiers MLP9, MLP117, MLP128, 3 combinations of two classifiers and one combination of three classifiers for the two combination rules. We must not forget that the aim of our system is to restrict the analyze of the whole document to a few fields likely to be numerical fields respecting a particular syntax. The most important criterion to evaluate such a method is therefore its capacity to detect all the fields of interest. A field is considered to be well detected if and only if no connected component in the labeled field is rejected and all the connected components in the detected field are included in the labeled field. Thus, we define the detection rate, calculated as follows:

$$DetectionRate = \frac{(nbofwelldetectedfields)}{(nboffields)}$$

Table 5 shows the detection rates (TOP1/ TOP2 / TOP5) on a test set of 293 images from incoming mail documents, containing 324 Zip codes, 289 phone numbers and 153 customer codes. A first observation is that the MLP128 and MLP117 provide much better detection results than KNN9 and MLP9. This observation confirms that we need intrinsic features to discriminate the four classes. We also remark that MLP9 provides poor results in comparison to KNN9. This is due to the fact that MLP is a parametric classifier, which modelizes the frontiers between classes. However, the reject class has complex frontiers as it can be considered as the negation of all the other classes. As we can see on these results, the combination of two or three classifiers generally provides better detection results than a simple classifier; which justify those combinations.

Concerning the combination rules, one can remark significant differences between product and mean. In our problem, product seems to be better than arithmetic mean, except for the combination of MLP117 and MLP128 classifiers. Finally, the best detection results are obtained with the MLP117 and MLP128, combined with the arithmetic mean, providing increases of respectively 21, 9 and 12 points in TOP1 compared with the original method KNN9. This combination is thus adopted.

Let us recall that our method aims at localizing – but not recognizing – the fields of interest in the document, in order to feed a recognizer. Thus, the goal is to reject the major part of the document, in order to process the recognition only on a few fields. According to the fact that our method is not able to detect all the fields of interest and only them, we prefer to localize too much fields with some false alarm. It means that some fields can be localized in the document whereas they are not numerical fields. The number of false alarm must be as low as possible. Thus, we define the false alarm rate,



DETECTION RATE(%)	ZIP codes	Phone numbers	Customer codes
KNN 9	49 / 64 / 80	65 / 74 / 79	57 / 70 / 77
MLP 9	40 / 56 / 77	56 / 68 / 75	56 / 67 / 72
MLP 117	61 / 71 / 83	66 / 73 / 81	63 / 76 /80
MLP 128	62 / 74 / 82	62 / 73 / 80	63 / 69 / 77
Prod (MLP 9 , MLP 117)	63 / 72 / 82	70 / 77 / 82	68 / 73 / 81
Mean (MLP 9 , MLP 117)	56 / 73 / 84	69 / 79 / 84	59 / 72 / 80
Prod (MLP 9 , MLP 128)	62 / 75 / 84	70 / 80 / 82	66 / 71 / 78
Mean (MLP 9 , MLP 128)	54 / 72 / 83	64 / 76 / 80	61 / 68 / 76
Prod (MLP 117 , MLP 128)	65 / 73 / 83	64 / 71 / 80	65 / 71 / 80
Mean (MLP 117 , MLP 128)	70 / 79 / 88	74 / 81 / 85	69 / 71 / 78
Prod (MLP 117 , MLP 128 , MLP 9)	62 / 72 / 81	62 / 70 / 75	60 / 71 / 80
Mean (MLP 117 , MLP 128 , MLP 9)	58 / 76 / 85	70 / 79 / 84	62 / 71 / 80

Tab 5: detection rate for ZIP codes, phone numbers and customer code

calculed as follows:

$$False\ alarm\ rate = 1 - \frac{nb\, of\ well\ detected\ fields}{nb\, of\ detected\ fields}$$

The false alarm rate gives us an indication about the ability to reject a major part of the document: for example, a 93% false alarm rate means that 7% of the fields detected by the method are real fields in the documents, whereas 93% are not fields of interest. The false alarm rates are presented on tab. 6.

FALSE ALARM RATE (%)	ZIP codes	Phone numbers	Client codes
KNN 9	93	77	89
Mean (MLP 117 , MLP 128)	89	77	88

Table 6: False alarm rates for ZIP codes, phone numbers and customer code

One can see that the false alarm rate has been slightly decreased by the use of combination of classifiers. An important precision is that some fields can be found in others : while processing the detection of zip codes (5 digits), it is obvious that some can be found in a phone number (10 digits, with eventually presence of separators) or in a customer code (8 digits with an optional separator between the first and the second digit).

5. Conclusion and future works

We have presented in this paper a new classification method to improve the approach presented in [10] for extracting numerical fields from handwritten incoming mail documents. Different families of feature sets are extracted: a contextual/morphological feature set, a chaincode feature set and a statistical/structural feature set. A multilayer perceptron is trained over each feature set, and a measurement-level combination of classifiers is achieved. The results on handwritten incoming mail documents show that the combination of these MLPs permits to benefit from the complementary of each feature set.

Our future works will focus on taking into account the triple digits. Indeed, some fields are not detected because they contain three touching digits. It must be taken into account in the fields syntax, but above all, we will have to correctly discriminate them. For that, a new feature set based on the water reservoir [17] could help us in this task. An other difficult problem is to detect the fields that on two lines: the syntactical model is not able to detect such numerical fields yet.

6. References

[1] C.Bishop, “Neural Network for Pattern Recognition” Oxford, New York: Clarendon Press; Oxford University Press. xvii, 482, 1995

[2] V. Di Lecce, G. Dimauro, A. Guerriero, S. Impedovo, G. Pirlo, A. Salzo, “Classifier Combination: the role of a-priori knowledge”, IWFHR VII, pp.143-152 , 2000.

[3] F. Grandidier, R. Sabourin, C.Y. Suen, and M. Gilloux, “A New Strategy for Improving Feature Sets in a Discrete HMM-Based Handwriting Recognition System”, IWFHR VII, pp. 113-122, 2000.

[4] A.F.R. Rahman, M.C. Fairhurst, “Multiple classifier decision combination strategies for character recognition: A review”, IJDAR, Vol. 5, pp.166-194, 2003.

[5] G.D. Forney, “The viterbi algorithm”, proc. of the IEEE 61, pp. 268-278, 1973.

[6] L. Heutte, P. Pereira, O. Bougeois, J. V. Moreau, B. Plessis, P. Courtellemont and Y. Lecourtier: “Multi-bank check recognition system: consideration on the numeral amount recognition module”, IJPRAI, Vool.11, pp. 595-618, 1997.

[7] L. Heutte, T. Paquet, J.V. Moreau, Y. Lecourtier, C. Olivier, “A structural / statistical feature based vector for handwritten character recognition”, Pattern Recognition Letters Vol. 19 pp. 629-641, 1998.

[8] G. Kim, V. Govindaraju, “A lexicon driven approach to handwritten word recognition for real time application”, IEEE Trans. on PAMI, Vol. 19, no. 4, pp.366-379, 1997.

[9] F. Kimura, S. Tsuruoka, Y. Miyake, M. Shridhar, “A lexicon directed algorithm for recognition of unconstrained handwritten words”, IEICE Trans. Inf.&Syst. Vol. E77, pp. 785-793, 1994.

[10] G. Koch, L. Heutte and T. Paquet, "Numerical sequence extraction in handwritten incoming mail documents", ICDAR , pp 369-373, 2003.

[11] S. Lawrence, I. Burns, A.D. Back, A.C. Tsoi and C. Lee Giles, “Neural Network Classification and Unequal Prior Class Probabilities”, in “Tricks of the Trade", Lecture Notes in Computer Science State-of-the-Art Surveys , pp. 299-314, 1998.

[12] Y. L. Cun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubband and L. D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Computation, pp. 541-551, 1989.

[13] L. Likforman-Sulem, C. Faure, “Une méthode de résolution des conflits d'alignements pour la segmentation des documents manuscrits”, CNED'94, Rouen, pp. 265-272, 1994.

[14] X. Liu and Z. Shi: “A Format-Driven Handwritten Word Recognition System”, ICDAR, pp. 1118-1122, 2003.

[15] S. Madhvanath, G. Kim, V. Govindaraju: “Chaincode Contour Processing for Handwritten Word Recognition”, Trans. PAMI pp. 928-932, 1999

[16] Morita, El Yacoubi, A. Bortolozzi, F. Sabourin, “Handwritten Month Word Recognition on Brazilian Bank Cheques” ICDAR, 2001.

[17] U. Pal, A. Belaid and Ch. Choisy, “Water Reservoir Based Approach for Touching Numeral Segmentation”, In Proc. Sixth ICDAR, pp 892-896, 2001.

[18] M.D. Richard and R.P. Lippmann : "Neural network classifiers estimate Bayesian a posteriori probabilities". Neural Computation, Vol. 3, pp. 461-483, 1991.

[19] O.D. Trier, A.K. Jain, T. Taxt, “Feature extraction methods for character recognition – A survey”, Pattern Recognition, Vol 29, no 4, pp. 641-662, 1996.

[20] L. Xu, A. Kryzak, C.Y. Suen, K. Liu, “Method of combining multiple classifiers and their applications to handwriting recognition”, IEEE Trans. on SMC, Vol. 22, no. 3, pp. 418-435, 1992.