

Hybrid OCR combination approach complemented by a specialized ICR applied on ancient documents

Hubert Cecotti and Abdel Belaïd
LORIA/CNRS Campus Scientifique BP 239,
54506 Vandoeuvre-les-Nancy cedex France
cecotti@loria.fr, abdel.belaid@loria.fr

Abstract

In spite of the improvement of Commercial Optical Character Recognition (OCR) during the last years, their ability to process different kinds of documents can also be a default. They cannot produce a perfect recognition for all documents. However they allow producing high result for standard cases. We propose in this paper a model combining several OCRs and a specialized ICR (Intelligent Character Recognition) based on a convolutional neural network to complement them. Instead of just performing several OCRs in parallel and applying a fusing rule of the results, a specialized neural network with an adaptive topology is added to complement the OCRs in function of the OCRs errors. This system has been tested on ancient documents containing old characters and old fonts not used in contemporary documents. The OCRs combination increases the recognition of about 3% whereas the ICR improves the recognition of rejected characters of more than 5%.

1. Introduction

Combining multiple classifiers has been recently a topic of great interest in pattern recognition and character recognition. It has been shown in the literature that several schemes can outperform the individual classifiers in order to increase the performance [1, 4, 10, 15]. To obtain an optimal classification system, several classifiers can be combined in a first step. However the global performance will depend on their complementary rate. Several problems occur to obtain the best result. Firstly, classifiers have to be chosen in order to complement each other's. This may not be an easy task as classifiers algorithm as their training data may be unknown. Secondly, the multi-classifier architecture has to be created. With a finite number of classifiers, the best combination method has to be found. This choice depends of

the number of classifiers, their behavior between themselves, the size of training data available for the combination... Several strategies are possible for designing a multi-classifier system. A first strategy consists in applying the classifiers in parallel: a single combination function merges the outputs of each individual classifier. In the second strategy scheme, the classifiers operate in cascade: classifiers are applied in succession, with each classifier producing a reduced set of possible classes. Each strategy has its drawback; the first strategy assumes complementary classifiers whereas the second assumes competitive classifiers. We propose a hybrid model where the first stage of the system is composed of classifiers connected in parallel whereas the second part is a specialized neural network specialized in finding patterns rejected in the first stage. This system has been tested on ancient documents. The classifiers of the first stage are commercial OCRs and the specialized classifier is a convolutional neural network, its topology being driven by the error of the first stage. In a first part, the different parts of the system will be described. Then the strategy used to extract and analyze errors as well as its exploitation for designing the neural network topology will be presented. In the third part we will describe the relationship between the OCRs combination results and the classifier specialized in rejection. Finally, the last part shows the recognition improvement given by our system.

2. System overview

This paper describes a character recognition system specifically tailored to ancient documents. Commercial OCRs are usually trained to recognize all kind of documents and are not specialized for one of them particularly. As a consequence, these generic OCR characteristics ensure a good performance on the majority of characters. They unavoidably lead, for a low proportion of documents, to a bad performance, as they are less fre-

quent or do not correspond to the trained character models. The use of OCRs for old printed documents is always impeded by the presence of some characters, which can not be not well recognized because of their unknown patterns or their deformations. OCRs act in different ways depending on the quality of the document. Thus their combination should give a better performance on classical characters. In order to continue to take advantage of OCRs performance on well-written characters, the first stage is completed by an additional ICR (Intelligent Character Recognition) capable of adapting its topology on the confusion errors. The error analysis is a very important task in qualifying the error type that should be processed further. It highlights rejected patterns during the test phase. The specific ICR is able to correct the error by specifying its own topology. The OCRs combination has two objectives:

- To enhance the OCR performances on common characters written in known font styles.
- To qualify the rejection errors among classical substitution, deletion, addition and segmentation errors.

3. Combination methods

Combination methods can be divided into three categories:

- Vertical combination schemes: serial combinations. Each classifier is performed sequentially. For example, each classifier is specialized on processing the rejected patterns of the previous classifier.
- Horizontal combination schemes: parallel combination. The classifiers work independently and concurrently. A fusion module combines their results.
- Hybrid combination schemes: it corresponds to the use of the two previous schemes, like the proposed system.

Several methods have been proposed so far in the literature. They can be generally classified into three main categories depending on the input of the fusing rule:

- Abstract level: use of the top candidate provided by each classifier.
- Ranked level: use of the entire ranked list of candidates.
- Measurement level: use of the confidence value of each candidate in the ranked list.

Several combination methods are described in the literature for fusing results: voting methods, Bayesian combination, Dempster-Shafer, behavior-knowledge space, neural networks, decision trees, etc. In the presented system, as the recognition algorithm and the training data used for

each OCR may not be available; the choice of the combination methods is limited. OCR outputs also give the confidence value of the first best choice and the probability of all the characters is unknown. The system is a hybrid combination. A horizontal topology scheme is used for the OCRs combination as the first step of the system. The relationship between the OCRs combination and the ICR follows a vertical scheme where the ICR processes rejected characters. For the OCRs combination, several methods have been tested: majority voting, behavior-knowledge space and neural networks allowing a double resolution of the label and the error type. The two first methods work on abstract level and the last one works on measurement level. We note LC_{m0} , the character list results of the combination.

3.1. Voting method

The majority voting method is an easy method to implement and he has shown good results in the literature [2, 6, 8]. The voting method assumes that to obtain good result, at least one classifier supports the character. The recognition is reached when all the classifiers support the character. This has as consequence to avoid correcting the common errors and leads to reject the maximum of errors instead of correcting them. Considering only two OCRs, the phenomena are accentuated, as there are less correction possibilities. Votes can be weighted with OCR knowledge. Although this method with only two OCRs may not always improve the recognition rate, it can though improve the reliability of the results.

3.2. Behavior-Knowledge Space

One of the strongest conditions to combine several classifiers based on conditional probability as formulated in the Bayes rule is that the classifiers must act independently. This condition is not easy to verify. That is why we preferred the use of the behavior-knowledge space (BKS) [5] method that makes no assumption about the classifier dependence. A behavior knowledge space is a D -dimensional space, each dimension corresponding to the decision of one classifier. Each classifier has as decision values the total number of classes N . Let $x \in C_i$ be the character to be recognized belonging to the class C_i . Let $s_k = j_k, k = 1..D$ be the k^{st} classifier among D and j_k its answer for the current character x . The probability that $x \in C_i$ is defined by the following formulae:

$$Belief(C_i) = \frac{P(s_1(x) = j_1, \dots, s_D(x) = j_D, x \in C_i)}{P(s_1(x) = j_1, \dots, s_D(x) = j_D)}$$

A cell of the BKS corresponds to the intersection of the individual classifiers decisions. Each point of the BKS is

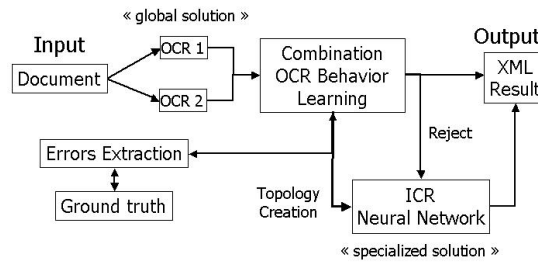


Figure 1. System overview.

noted by $BKS(j_1, \dots, j_D)$, $j_i = 1..N$; and contains a vector of size N : $bks(j_1, \dots, j_D)(i)$, $i = 1..N$.

Let $bks(j_1, \dots, j_D)(i)$ be the total number of characters x such that $s_1(x) = j_1, \dots, s_D(x) = j_D$ and $x \in C_i$, $i = 1..N$. Let $T(j_1, \dots, j_D)(i)$ be the total number of characters x such that $s_1(x) = j_1, \dots, s_D(x) = j_D$. The best representative class of $BKS(j_1, \dots, j_D)$: R is defined by:

$$R = \operatorname{argmax}(bks(j_1, \dots, j_D)(i)), i = 1..N$$

If one cell of the BKS is empty then the pattern is naturally rejected. A small database may be a problem in obtaining a good generalization and many empty cells may occur if the database is not representative. As the BKS size increases exponentially with the number of classifiers, the data sets has to increase in the same way [11]. For BKS cells where the most representative class is defined by a low probability, meaning ambiguous cases, characters are rejected. R is rejected if $\operatorname{Belief}(C_i) \leq \alpha$ where α is a threshold representing the recognition quality desired.

3.3. Neural Network

Neural networks can also achieve such combination, and some neural network models have already been applied for characters recognition [14]. Our approach with the neural network consists in extracting simultaneously the label and the error type. The neural network in our experiments takes as input two vectors V_α and V_{error} . V_α is the confusion vector of each vocabulary character, whereas V_{error} is the error status of the character according to the error types: substitution, deletion, addition, etc. The network is able to perform a simultaneous analysis of the error and the label, and it can also allow a better generalization than the BKS method. The neural network can solve some generalization problem of the BKS empty cells. However the learning phase of the neural network as the testing phase are slower than the BKS ones.

4. Errors extraction and categorization

After the combination process, errors have to be extracted properly to specify the ICR. The OCRs combination complements and solves the ambiguous cases; the ICR needs the OCRs combination errors to adapt its topology. This analysis is obtained by differentiating the OCRs results and a ground truth of the document. The error analysis of the differentiation of the OCRs combination or the individual OCR results will provide all the information needed to create the specialized ICR, directly complementary to the combination. OCRs commit several types of errors as: confusion, addition, deletion or segmentation. The ICR will employ these errors in order to adapt its topology. Error types are detected by a comparison between two character lists: L_{GT} representing the ground truth and LC_{m0} obtained by OCRs combination. An appropriate dynamic programming algorithm is used to optimize the alignment of the lists. Let A, B, C, D be characters and let $\#$ be the alignment character, the errors can be categorized as expressed in the following rules:

Confusion: $A \leftarrow B$

Addition: $\# \leftarrow B$

Deletion: $A \leftarrow \#$

Fusion: $AB \leftarrow C\#$

Cutting: $A\# \leftarrow CD$

These errors correspond to rules extracted by using the edit distance. This edit distance was proposed by Seni, Kripasundar and Srihari, which is an improved version of the Damerau-Levenshtein metric [3, 13]. The error types mentioned below are easy to determine when the error chains are small, usually occurring in clean documents. Inversely, the errors meaning are very difficult to locate when the erroneous chains are large (i.e. occurring in complex documents). The error covers several contiguous characters making the error type difficult to determine, as the correspondence between characters is not obvious. The alignment problem is transforming in locating the error origin in this

long chain? (i.e. what is the rule for each character responsible of this error?). The errors are located recursively based on the erroneous chain lengths, starting from the small errors to detect the biggest ones. The procedure starts by locating the small erroneous chains in the entire document. If one of the found error occurs in the largest erroneous chain, this chain is divided in two parts: prefix before the known error, and suffix, after the error, which are both recursively analyzed according to other smaller errors detected in the document. It is obvious that this approach can work only when the erroneous chain length is reasonably large. Errors that cannot be analyzed properly are ignored for the estimation. For example, if a very noisy textual part of the document is recognized as an image, there will be a long erroneous chain corresponding to a deletion. However, this deletion is produced by a global mistake of the OCRs and does not correspond to the real OCRs behavior for each letters of the chain. Once the errors are detected, we generate for each one a probability function defined by: M_r : confusion matrix, M_f : fusion matrix, M_s : cutting matrix, V_a : addition vector, V_d deletion vector. For a character x recognized as belonging the class i , if the recognition rate of the class is greater than a threshold representing the desired quality of the document, then the character is accepted. Otherwise the image of the characters and the different confusion classes of i are given as input of the ICR.

5. Error correction

We use a specialized classifier, called ICR, for error correction, acting directly on the image pixels of the rejected characters. This ICR is a modified multi-layer Perceptron with convolutional layers [7]. The neural network is composed of 5 layers:

- The first one corresponds to the input image, normalized by its centering.
- The next two layers corresponds to the information extraction, performed by convolutions layers using weight sharing.
- The fourth layer is composed of neurons pool, each pool being specialized for a class. The links between the fourth and the last layers are function of the error previously detected.
- The last one corresponds to the output with a number of neurons equal to the total number of classes.

Characters rejected by the OCRs combination or labeled as erroneous characters are treated by the ICR. The confusion is the error that can be performed by the ICR. However, as we do not know if the image corresponds to a character image or to a character portion occurred as a result of a segmentation problem, the confusion matrix M_r is weighted by

the addition and deletion vectors. The remaining errors are not integrated because they do not directly step at the character level (i.e. image level) but they allow to better qualifying the confusion measures.

The erroneous character image is taken into account by the ICR if and only if the maximum of the confusion rate is lower than a fixed threshold S (i.e. $\max_i(M_r(i, C_{m0})) < \alpha$). Considering the image and its associated result j . In the fourth and last layers, for each neurons pool k , they are weighted by $M_r(j, k)$, $k = 1..N$. If $M_r(j, k) = 0$, which is equivalent to the absence of links for the k^{th} pool. The ICR do not work alone for the rejected patterns but uses the OCR behavior knowledge. The relationship between the OCR and the ICR are then mutual. The ICR performs OCR rejected characters and the ICR needs the OCRs errors to build its own topology.

6. Experiments

The system has been tested on ancient documents. This kind of documents are real challenge to OCRs [12, 9]. These documents are pages from a French dictionary of the XVIII century: "dictionnaire de Trevoux" containing very special characters not used anymore, so naturally disturbing the OCRs. For example, the letter "s" can be written in two shapes: the standard "s" and the long "s". In this case the OCR has a confusion problem between "s" and "f", then the neural network specializes its topology to differentiate shapes of "s" and "f" to reduce the confusion. In this case, the neural network is a tool used not only to solve ambiguous case, but also to differentiate classes. The database is composed of 8 pages of the dictionary chosen to be the best representatives. The pages are scanned at 300dpi and were binarized. Half of the documents are used for training and the other half is used for testing. The results obtained just by combining 2 OCRs are shown in table 1. We note that BKS and the neural network (NN) give the best improvements. In the rejected pattern and the confusion classes, the main errors are due to noise, the presence of accents, or to new classes not found by OCR. The more representative error for these documents is the confusion between the characters "f" and "long s", which have almost the same shape. Figure 2 presents some special characters and ligature characters with their corresponding in Arial font. If we consider the 4 classes: "f,i,l,s" the first OCR has 91.03% and the second has 93.27%. The two "s" are considered as the same character during the combination. With the OCR, we obtain 99.09% of recognition for the 4 previous classes, and it solves the ambiguity between the 2 "s". The table 2 presents the recognition rate obtained for the all the rejected patterns with and without the adaptive topology (i.e with and without the OCRs errors knowledge).

f	ct	st	sh	ss	ssi	sf
s	ct	st	sh	ss	ssi	sf

f	ct	st	sh	ss	ssi	sf
s	ct	st	sh	ss	ssi	sf

Figure 2. Peculiar characters.

Method	Recognition	Rejection	Error
OCR 1	85.14	0.06	14.80
OCR 2	87.28	0.04	12.68
Vote	77.28	11.22	11.50
BKS (train)	92.64	0.02	7.34
BKS (test)	88.48	0.01	11.51
NN (train)	90.24	0.0	9.76
NN (test)	88.18	0.0	11.82

Table 1. OCR alone and with combination.

7. Conclusion

We have presented a hybrid multi-classifier model using a specialized neural network for rejection processing. The system has been applied on ancient documents and several fusion methods have been compared. This approach has been successful in several ways. The recognition rate has improved and the ambiguous cases have been highlighted, thanks to the error analysis. The ICR based on a convolutional neural network complements the OCR thanks to its topology, and allows solving some ambiguous cases. In some cases, like ancient document recognition, fusion rule are not enough due to the classifier behaviors. It becomes then necessary to use complementary tools. The system might be improved by adding new OCRs but it would be more difficult to extract complementarities between them as they already work very well.

8. Acknowledgements

We wish to thank the ATILF laboratory (*Analyse et Traitement Informatique de la Langue Francaise*) and especially Prof. J.-M. Pierrel and I. Turcan for providing the input images.

	Classical Topology	Adaptive Topology
Train	96.63	99.28
Test	93.59	99.27

Table 2. Recognition rate for ambiguous characters.

References

- [1] D. Bahler and L. Navarro. Methods for combining heterogeneous sets of classifiers. In *17th Natl. Conf. on Artificial Intelligence (AAAI 2000), Workshop on New Research Problems for Machine Learning.*, 2000.
- [2] A. Belaid and J. Anigbogu. Use of many classifiers for mult-font text recognition. In *Traitement du signal, vol. 11, no. 1, pp. 57-75*, 1994.
- [3] F. Damereau. A technique for computer detection and correction of spelling errors. In *Communications of the ACM, vol. 7, pp. 649-664*, 1964.
- [4] V. Gunes, M. Menard, P. Loonis, and S. Petit-Renaud. Systems of classifiers: state of the art and trends. In *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), 17(8), World-Scientific.*, 2004.
- [5] Y. Huang and C. Suen. A method of combining multiple experts for the recognition of unconstrained handwritten numerals. In *IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 17, no. 1, pp. 90-94*, 1995.
- [6] L. Lam and C. Suen. Application of majority voting to pattern recognition: an analysis of its behavior and performance. In *IEEE Trans Pattern Anal Mach Intell, vol. 27, no. 5, 1997, pp. 553-568*, 1997.
- [7] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324.*, 1998.
- [8] D. Lopresti and J. Zhou. Using consensus sequence voting to correct ocr errors. In *Computer Vision and Image Understanding, vol. 67, no. 1, pp. 39-47*, 1997.
- [9] T. Nartker, K. Taghva, R. Young, J. Borsack, and A. Condit. Ocr correction based on document level knowledge. In *In Proc. IS&T/SPIE 2003 Intl. Symp. on Electronic Imaging Science and Technology, vol. 5010, pp. 103-110*, 2003.
- [10] A. Rahman and M. Fairhurst. Multiple classifier decision combination strategies for character recognition: A review. In *International Journal on Document Analysis and Recognition (IJ DAR) vol. 5, pp. 166-194*, 2003.
- [11] S. Raudys and F. Roli. The behavior knowledge space fusion method: Analysis of generalization error and strategies for performance improvement. In *Multiple Classifier Systems, 4th International Workshop, MCS 2003, pp. 55-64*, 2003.
- [12] C. Ribeiro, J. Gil, J. C. Pinto, and J. Sousa. Ancient document recognition using fuzzy methods. In *Proc. of the 4th International Workshop on Pattern Recognition in Informations Systems, pp. 98-107*, 2004.
- [13] G. Seni, V. Kripasundar, and R. Srihari. Generalizing edit distance for handwritten text recognition. In *In Proceedings of SPIE/IS&T Conference on Document Recognition, pp. 54-65, San Jose, CA, 1995*.
- [14] A. Sharkey. Combining artificial neural nets: ensemble and modular multi-net systems. In *Perspectives in neural computing. Springer, Berlin Heidelberg New York*, 1999.
- [15] C. Suen and L. Lam. Multiple classifier combination methodologies for different output levels. In *Springer-Verlag Pub., Lectures Notes in Computer Science, Vol. 1857 (J.Kittler and F.Roli Eds.) pp. 52-66.*, 2000.