2014 14th International Conference on Frontiers in Handwriting Recognition

A Comparison of Recognition Strategies for Printed/Handwritten Composite Documents

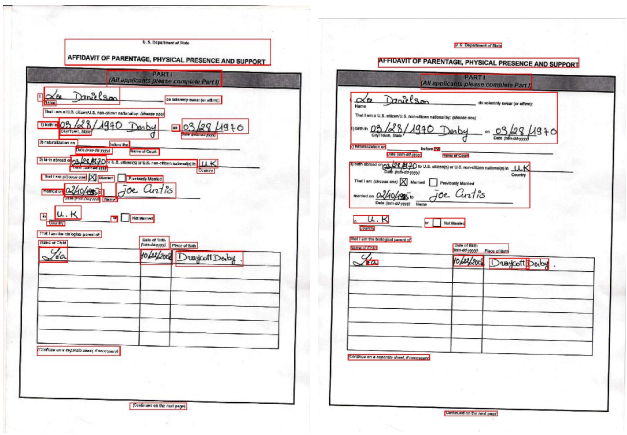
Bastien Moysset, Ronaldo Messina, Christopher Kermorvant
A2iA, 39 rue de la Bienfaisance, 75008 - Paris - France

***Abstract*—Full-page segmentation and recognition of real-world documents is a challenging task, involving the segmentation of the images (graphics, text) and the subsequent recognition of the detected text-zones. Often those documents present zones with both write-types: printed and handwritten, which so far have been dealt with by classifying the zones according to the write-type and then using type-specific models for recognition.**

Here we present two recognition systems using state-of-the-art recurrent neural networks, that can recognize the text in zones with both write-types, without the need of explicit type identification; just the segmentation in lines is needed. In one of the systems, there is no distinction on the type at the network's output (one output label per character) while in the other there is one output label for each character and write-type.

Experiments have been done on real-world documents from the Maurdor competition. These two systems perform at a similar level than systems using specific networks per type on the constrained task where there is only one write-type per zone. They perform better when both handwritten and printed text are present in the text zone.

The results open the perspective to treat OCR and handwritten text recognition with a single optical model.



(a) Ground-truth annotation of text zone. (b) Automatic text zone detection.

Fig. 1: A comparison of the ground-truth for text zone location with automatically detected text paragraphs for a document mixed printed and handwritten zones.

I. Introduction

The objective of text recognition, printed or handwritten, is to transcribe text from images of documents for information retrieval, for example: content-based document classification, named entities extraction or writer identification. If the performance of OCR for printed text on clean documents, such as good quality scans of books, is well demonstrated, the ability of the systems to deal with noisy and complex documents, containing both printed and handwritten text has yet to be evaluated. The Maurdor evaluation campaign [1] targeted this evaluation: considering a complete document processing workflow, what is the performance of the complete systems and of each step on a large variety of complex multi-lingual documents. The complete document processing chain was decomposed into autonomous modules: document layout analysis, write-type identification, language identification, text recognition, logical organization and information extraction. Each module was evaluated in isolation with the ground-truth value of the data from the previous module in the sequence. For example, the text recognition modules were evaluated using as input the co-ordinates of the text zones, the write-type of the text and the language. The advantage of the evaluation setup is that each step in the processing chain is evaluated in a very controlled setting. The drawback is that the definition of the input/output of the modules are interdependent: if the recognition module has to recognized homogeneous text of a certain type, for example printed French text, the Document Layout Analysis (DLA) module need to split the text zones into homogeneous zones of this type. This problem is illustrated on Figure 1. The text zones provided to the test recognizer contains exclusively printed or handwritten text, as it is supposed that the previous modules were able to separate and detect the two writing types. However, in real systems, DLA module usually provides text zones that are not homogeneous. In this paper, we present an analysis of the impact of mixed text paragraphs on recognition and propose to train mixed write-type text recognition systems to tackle the problem.

Few systems were proposed to deal with both handwritten and printed text at character level [2] [3] [4], especially for digit recognition. But, to the best of our knowledge, no study was performed on the feasibility of an efficient system able to recognize *both* write-types (printed and handwritten) at line level.

In this paper, we describe two systems trained to deal with mixed texts for full-line recognition. These systems are based on the Recurrent Neural Network (RNN) system used during the Maurdor competition [5]. We present on the one hand a system in which printed and handwritten characters share the same network labels and, on the other hand, a system in which the different write-types have distinct output labels. These systems are compared to systems dedicated to handwritten text recognition (HWR) or printed text recognition (PRN).

2167-6445/14 $31.00 © 2014 IEEE
DOI 10.1109/ICFHR.2014.34

TABLE I: Number of document images, line snippets and words in all the sets (Train, Dev1, Dev2 and Test) for printed and handwritten text.

		PRN	HWR
Train	#Images	1608	1003
	#LineSnippets	49210	10825
	#Words	247384	28641
Dev1	#Images	144	95
	#LineSnippets	4401	1115
	#Words	25682	2809
Dev2	#Images	164	95
	#LineSnippets	5413	969
	#Words	27295	2019
test-EN-only	#Images	242	
	#Words	59212	5501
test-EN-only-C1	#Images	32	
	#Words	12895	1779

The paper is divided as follows. Section II describes the database on which was performed our study. We present the experimental setup in Section III where we also describe the proposed systems. The results and analysis of our experiments are presented in Section IV. We propose in Section V some perspectives for further improvements, before concluding in Section VI.

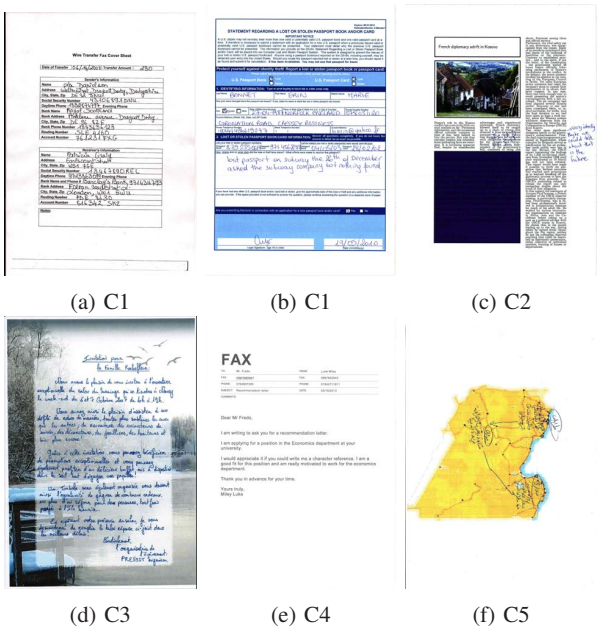
II. The Maurdor Database

We have tested the different strategies for mixed text recognition on the documents from the Maurdor database. This database is composed of 8774 pages containing printed and handwritten text in French, English and Arabic. The database is fully annotated with the coordinates of text zones, the write-type (printed or handwritten), the language and the textual content. The annotators were instructed to delimit the text zones containing only one type of text (printed or handwritten), but the zone could contain several lines of text. We used an automatic procedure [6] to produce the annotations (coordinates and textual context) of all the text lines. The official splits (Train set, Dev set and Test set) were used but, in this study, we selected the documents containing only English text for evaluation. The development set was split into two parts (Dev1 and Dev2) for the optimization of the hyper-parameters of the recognition system.

The documents of the Maurdor database are also classified into one of the following categories, shown on Figure 2: printed forms with handwritten information (C1), commercial documents (C2), handwritten private correspondance (C3), typed and handwritten private or professional correspondance (C4), Other kinds of documents (C5). We present a detailed analysis of the results on the documents from the C1 category as the text zones usually contain both printed and handwritten text. A summary of the data statistics is presented on Table I.

III. Full Page Mixed Text Recognition system

In this section, we present the different modules composing a full page text recognition system for complex documents with both printed and handwritten texts. Figure 3 shows the complete system and the different strategies evaluated in this paper. We have at disposal several automatic algorithms at each step and compare them to the ground-truth when available. Note that the main goal is to test the possibility of doing no write type detection.



(a) C1 (b) C1 (c) C2

(d) C3 (e) C4 (f) C5

Fig. 2: Samples of documents from the Maurdor database, for each of the five document categories.

A. Text zone detection

This modules analyses a complete page image to detect and type all the graphical objects: text zones, graphical zone, tables, images, separators, signatures, etc. This module takes as input a document image and outputs the coordinates of the different zones with their write-type. In this study, we are only interested in the text zones, which can be composed of one or several lines of text. We have have evaluated the text recognition on top of two strategies, using either:

a) the ground-truth coordinates of the text zones: provided in the Maurdor evaluation data;

b) the text zone detector developed by the LITIS laboratory (Rouen, France): this systems separates text and non-text elements using a MLP based on features related to the shape of the connected components [7].

B. Writing-type detection

The writing-type detection module predicts if a text zone contains printed or handwritten text. In the Maurdor evaluation setup, the text zones were supposed to contain only one write-type of text: printed or handwritten, exclusively. This was the case for the ground-truth, but not for the text zones coming from an automatic detector such as the LITIS system presented in the previous section.

We have evaluated the impact of thee different write-type detector on the complete system, using either:

a) the ground-truth write-type value of the text zones: provided in the Maurdor evaluation data;

b) the detector developed by the IRISA lab (Rennes, France): it first segments the paragraph into words and then

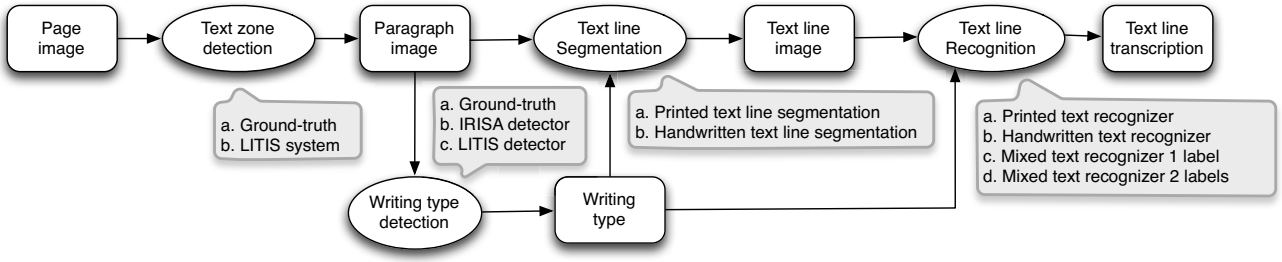


Fig. 3: Different strategies for the text recognition process.

classifies each words into handwritten or printed with a low-depth boosted decision tree trained with around 250 image-based features. The write-types of all the word are combined to predict the write-type of the text zone [8];

c) the detector developed by the LITIS lab (Rouen, France): it is based on an analysis of the connected components contours in the text blocks. Fixed length fragments of the contours are associated to fragments in a codebook for each type (handwritten, printed) and script (Arabic, Latin). A multi-layer perceptron classifies each connected component into a class and a vote gives the write-type of the paragraph [7].

During the second Maurdor campaign, both automatic detectors reached F-measures between 90% and 95% for write-type detection.

C. Text line segmentation

The text zone detected by the previous modules may contain one or several lines of text. For the recognition, each text line in this zone must be extracted. We have used two text line segmentation algorithms, depending on the writing-type.

a) On the handwritten zones: we used an algorithm based on grouping of connected components. Connected components are extracted from the binarized image after denoising, deskewing and deslanting. Based on their skeleton, the connected components are grouped into words and text lines based on statistical heuristics.

b) On the printed zones: the image is pre-processed with binarization, deskewing and cleaning to remove lines and noise. Text lines are then located by projection profile. A post-processing merges lines lying at the same level.

D. Optical model

Our optical model is based on a 2D multidirectional Recurrent Neural Network (RNN) [9] . Our implementation of this model scans the image in four different directions and produces predictions for each character in the given label alphabet. Details on the architecture and on the training of the model are given in [5].

For this study, we trained four different models:

a) A model specialized in printed recognition (RNN-PRN): this model is trained only on examples of printed text from the train set. Its character error rate on Dev1 is 1.8%;

b) A model specialized in handwriting recognition (RNN-HWR): this model is trained only on examples of handwritten text from the train set. Its character error rate on Dev1 is 15.5%;

c) A model for mixed printed/handwritten text with common output (Mix. RNN 1 label): this model is trained on examples of both handwritten and printed text from the train set. There is only one output for each character, that can be printed or handwritten. Its character error rate on Dev1 is 5.6%;

d) A model for mixed printed/handwritten text with separate label output for each type (Mix. RNN 2 labels): this model is trained on examples of both handwritten and printed text from the train set, in which the text lines are annotated accordingly. In this case, the last LSTM [10] layer before the soft-max has double the number of neurons than the network with single labels in order to cope with the increased number of labels at the output. Its character error rate on Dev1 is 3.3%;

E. Lexicon and language model

We use the Kaldi [11] toolkit, based on weighted finite-state-transducers (WFST) to find the best word hypothesis from the RNN predictions, computed on text line snippets.

These predictions are decoded using a composed FST ("HCLG" as in the usual recipe [12], which represents the search space for the valid word hypotheses) where "G" is a trigram language model and "L" is a lexicon transducer which puts the characters in relation to the words. The "HC" transducer is of no concern here and simply represent the RNN predictions as pseudo-likelihoods under the formalism of Hidden Markov Models.

The lexicon transducer is modified for the RNN with distinct output labels for each character type, so it does not matter if the word is written in printed or handwritten characters as shown in Figure 4 for the word "lexica"; It is allowed to change the write-type within a word because some characters are quite similar and the predictions could be high for either type. We could make the grammar output indicate the type of the characters by creating separate paths for each type, thus having a handwritten and a printed version of each word.

The language model used in the experiments is a word trigram estimated on all the data in the "Train" set, without

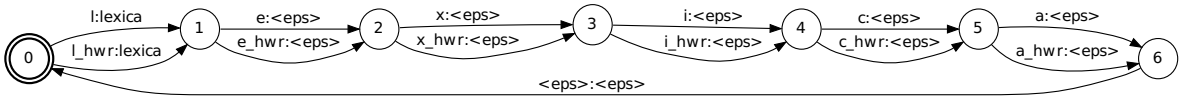


Fig. 4: Exemple on how characters are related to words in the lexicon.

distinction on the type (PRN or HWR). It was not pruned and Witten-Bell smoothing [13] was used.

Textual data was tokenized according to the rules found in the evaluation tool [1]. We treated punctuation characters as "word" tokens and split the digit strings to reduce vocabulary size; the inter-word space is a "word" token in the language model, it is related to the optical model by its associated output. The combination of modeling the inter-word spaces and the digits in isolation enables the recognition of any digit string, but it relies on (good) predictions from the optical model.

Having the inter-word space in the ngrams reduces the syntactical constraints on the words (we would need pentagrams to have the same syntactical constraints as usual trigrams without the inter-word space), but as the training data is relatively small we decide to stay at trigram level (an informal test with pentagram showed a slight improvement, but at the expense of a much larger decoding FST).

IV. Evaluation and Results

A. Evaluation method

The evaluation of the different strategies was based on the word error rate (WER) computed with the tool developed by the LNE for the Maurdor evaluation campaign [1]. For the experiments with automatic extraction of text zones (cf. Section IV-D), an alignment between the reference and hypothesis boxes is made before the computation of the word error rate. Reference and hypothesis blocks that are overlapping are grouped. If there are several reference boxes or several hypothesis boxes in a group, they are ordered using heuristics. A box comes before another one if it is completely above or if it is above and there is an important horizontal overlap. If none of these conditions are met, boxes are ordered from left to right.

B. Comparison of mixed systems and specialized systems on single write-type zones

We assess the performance of the neural networks trained on single write-type and of the neural networks trained on both write-types. It was not possible to train directly on mixed type snippets because the training data was annotated in paragraphs of separate write-types, even if the text was in the same "line". Decoding is also performed on paragraphs of a single type, but using either a RNN trained on the same type or RNNs trained on both types. We also decoded in a mismatched condition: the RNN trained on handwritten (resp. printed) type was used to recognize printed (resp. handwritten) text.

TABLE II: Comparison of word error rates (WER) for the different neural networks.

Recognition system	PRN	HWR	Average
RNN-HWR	62.2%	39.5%	60.3%
RNN-PRN	28.2%	94.4%	33.8%
RNN matched conditions	28.2%	39.5%	29.2%
Mix RNN, 1 label per character	29.7%	45.1%	31.0%
Mix RNN, 2 labels per character	30.7%	45.1%	31.9%

Results shown on Table II indicate that a single network can learn both of the writing-types. Having one output label makes it easier to deal with mixed-type lines, while having two labels makes type-detection possible. Depending on the application, either modeling method could be used with a slight degradation (maybe training with more balanced data would alleviate this) but both avoid having to explicitly detect the write-type. We can imagine that when training with snippets of two types, the network is presented with the hard task of modeling quite variable versions of each character into the same output label, while having separate labels for the characters makes it hard because there is the double number of output labels for the same amount of training data.

The best performance is obtained in the idealized conditions when a RNN trained only on the same write-type of character is used, namely "RNN matched conditions", i.e. using the RNN trained on handwritten (resp. printed) snippets on the handwritten (resp. printed) zones. This implies in having perfect location and detection of the write-type in the paragraph (a line segmentation that was optimized for each type was used). It is obvious that the RNNs trained on a single type yields low performance on the mismatched condition (for exemple, handwritten model on printed snippets).

The performance of the mixed RNN (either 1 or 2 labels per character) is close from the ideal situation where the write-type is known; it is about 5% relative lower for printed text and 14% for handwritten text.

As there is much more printed text in the training data, we can conjecture that the RNN just learned a better representation for printed text, as it was seeking to minimize the alignement error for that part of the data. Table I shows that the number of training snippets for printed text is about 4.5 times higher than handwritten, while the number of words is about 8.4 times larger.

TABLE III: Word error rates on the test set for the complete system with ground-truth text location and different write-type detection methods

RNN used	Type detector	PRN	HWR	Avg.
Matched type RNN	ground-truth	28.2%	39.5%	29.2%
Matched type RNN	IRISA	28.4%	46.0%	29.9%
Matched type RNN	LITIS	29.6%	43.6%	30.8%
Mix RNN, 1 label per character	LITIS	29.7%	45.9%	31.1%
Mix RNN, 1 label per character	IRISA	29.7%	45.7%	31.1%
Mix RNN, 2 labels per character	LITIS	30.7%	46.2%	32.0%
Mix RNN, 2 labels per character	IRISA	30.7%	46.1%	32.0%

C. Comparison between Mixed systems and Specialized systems coupled to write-type detectors

The results in Table II show that write-type detection can be an issue if the wrong optical model is used (using the printed model on handwritten images seeming to be the worst). Here we compare the performance of the mixed systems to systems using the write-type detectors from other laboratories participating in the Maurdor campaign as described in Section III-B. This study is made in the constrained case where we know there is only one write-type per zone. The results can be found in Table III.

Note that two results are given for each mixed RNN. In order to remove biases, the write-type detection results were used to choose the line segmenter either when using a mixed RNN or not.

The performance of the write-type detectors is quite good: between 95.6 and 97.6 on the F-measure (harmonic mean between precision and recall) for printed and between 86.4 and 91.5 for the handwritten part; the recognition results for the matched conditions are quite close to the results when the type is given. We can also see that the results for the two line detector algorithms with matched models seem to be better for one of the types, so a combination is plausible, so as to improve the detection.

Mixed type RNNs have performance close on average to the systems with write-type detection.

D. Comparison of systems on detected and potentially mixed paragraphs

Table IV presents results for automatically detected paragraphs, along with its automatically detected write-type. These data come from the work of the LITIS during the Maurdor campaign and are described in Sections III-B and III-A. In case the system was not sure about the type of the paragraph, it was assumed to be handwritten because the handwritten system was supposed to generalize better. The reference WER (i.e. the best results we could obtain in this set, with matched RNNs and when both the paragraph and write-type are given by the annotations) is 29.2% as seen in Table II.

The high word error rates can be explained by the accumulation of errors during the whole pipeline. Indeed, errors can occurs during the block segmentation, during the classification as text or non-text, during the write-type detection or because of the optical model. The paragraph detection step is the main source of error and explain a deletion rate of about 60% for all optical models and the 12-14% range for insertions. This cause the high word error rates in this section.

Models have also been evaluated on the C1 subset, which contains filled forms, and is more likely to present lines with both write-types, as illustrated in Figure 1. The recognition results are presented in Table IV using the paragraphs and write-type automatically detected using the LITIS algorithm; the reference result is 8.6%, when the type and paragraph are given by the annotations (and 11.3% for mixed networks).

These experiments show that when the paragraphs or the lines possibly contain both write-types, mixed neural networks give better results. This is especially true for the C1 subset where there are many paragraphs with both handwritten and printed text.

V. Perspectives

In the following we present some perspectives for this work:

- Train RNN with mixed lines: so far, our models were trained only on pure printed or handwritten snippets. It would be interesting to train them on snippets with mixed write-types, as this may have an influence due to the recurrent nature of the optical models we use.
- Multi-language: Languages that use Latin characters could be all learned by a single RNN, the main differences are some accentuated characters and some other diacritics. A challenge would be to learn character labels for mixed charsets, such as Latin and Arabic.
- Tune network architecture: Increasing the number of output labels should have an impact on the number of neurons in the intermediary layers, to allow more complex internal representations to be created.
- More balanced training data: There was much more printed data to train the RNNs; we can think that mixed models would do much better if the data was more balanced between printed/handwritten types, specially if the training is done on single type snippets. There are two “cheap” ways of doing so: one is to use the handwritten data from the French part of the corpus, but ignoring the accents on the characters (vowels plus the cedilla), and the other is to add noise and distortions to the available snippets, so as the network is presented to an equitable number of printed and handwritten samples.
- Customized language modeling: In the case of having separate labels for printed and handwritten characters, it is possible to make language models that are specialized, for example, in form fields. This avoids the combination of the outputs of two models (in the usual situation) and also increases the constraints on the predictions, making the modeling more precise and hopefully with less errors.

VI. Conclusion

In this article was studied the possibility of using a unique recurent neural network optical model for recognizing both

TABLE IV: Word error rates on the complete test set and the test set restricted to C1 documents for the complete system with automatic text location and different type detection methods

RNN used	Text detector	Write-type detector	WER	Del	Ins	Sub
Complete Test set						
matched type RNN	ground-truth	ground-truth	29.2%	21.6%	1.8%	5.8%
matched type RNN	LITIS	LITIS	83.9%	59.9%	13.7%	10.5%
Mix RNN, 1 label per character	LITIS	LITIS	83.5%	59.6%	13.5%	10.4%
Mix RNN, 2 labels per character	LITIS	LITIS	82.7%	60.3%	11.9%	10.5%
Test set restricted to C1 documents						
matched type RNN	ground-truth	ground-truth	8.6%	2.3%	1.5%	4.8%
matched type RNN	LITIS	LITIS	47.2%	20.7%	16.0%	10.5%
Mix RNN, 1 label per character	LITIS	LITIS	45.7%	19.2%	16.4%	10.2%
Mix RNN, 2 labels per character	LITIS	LITIS	45.0%	20.0%	14.6%	10.4%

handwritten and printed texts. Two strategies were experimented: the first with a single output label per character, the second with different outputs for printed and handwritten characters; they performed with similar results. Our experiments, carried out on the Maurdor database, have shown that neural networks trained only on data of one write-type: printed or handwritten in combination with a write-type detector, were slightly better in the constrained case where the paragraphs were of constant write-type. On the contrary, in the more realistic situation where both handwritten and printed text are present in the paragraph image, mixed-type neural networks gave better results. Moreover, with mixed-type neural models, a single system has to be trained. The cost of annotation is reduced because the write-type of the text zones is not needed anymore. The neural network with two outputs per character may also give information about the write-type at the character level (but annotated data for the character write-type is required for training).

Correct detection of paragraphs and lines remains quite challenging as shown by the worse results when automatic segmentation is used (c.f. Table IV). When the annotations provide the paragraph locations (lines were automatically detected) the performance was quite high, as shown in Table III. We expect to improve the performance of our mixed system with the work presented in the perspectives.

Using mixed-type models is an interesting alternative because of the simplicity those models can bring to the whole recognition pipeline and because it simplifies data annotation while showing good results.

Correct detection of paragraphs and lines is, of course, of paramount importance for the text recognition. We expect that avoiding the need of write-type detection due to mixed models would alleviate the task.

ACKNOWLEDGMENT

We would like to thank IRISA and LITIS for letting us use their write-type detection and segmentations results in our experiments. Those segmentations were helpful in assessing the interest of mixed-type modeling and also in indicating the importance in investigating on segmentation methods.

This work was partially funded by the French Defense Agency (DGA) through the Maurdor research contract with Airbus Defense and Space (Cassidian) and supported by the French Grand Emprunt-Investissements d'Avenir program through the PACTE project.

REFERENCES

[1] I. Oparin, J. Kahn, and O. Galibert, "First Maurdor 2013 Evaluation Campaign in Scanned Document Image Processing," in *International Conference on Acoustics, Speech, and Signal Processing*, 2014.

[2] J. Denker, I. Guyon, and Y. LeCun, "Time delay neural network for printed and cursive handwritten character recognition," Apr. 14 1992, uS Patent 5,105,468. [Online]. Available: http://www.google.fr/patents/US5105468

[3] D. C. Álvarez, F. M. Rodríguez, and X. F. Hermida, "Printed and Handwritten Digits Recognition Using Neural Networks."

[4] A. Majumdar and B. B. Chaudhuri, "A MLP Classifier for Both Printed and Handwritten Bangla Numeral Recognition." in *Indian Conference on Computer Vision, Graphics and Image Processing*, 2006.

[5] B. Moysset, T. Bluche, M. Knibbe, M.-F. Benzeghiba, R. Messina, J. Louradour, and C. Kermorvant, "The A2iA Multi-lingual Text Recognition System at the Maurdor Evaluation," in *International Conference on Frontiers of Handwriting Recognition*, 2014.

[6] T. Bluche, B. Moysset, and C. Kermorvant, "Automatic Line Segmentation and Ground-Truth Alignment of Handwritten Documents," in *International Conference on Frontiers of Handwriting Recognition*, 2014.

[7] P. Barlas, S. Adam, C. Chatelain, and T. .Paquet, "A typed and handwritten text block segmentation system for heterogeneous and complex documents," in *International Workshop on Document Analysis Systems*, 2014.

[8] Y. Ricquebourg, C. Raymond, B. Poirriez, A. Lemaitre, and B. Coüasnon, "Boosting bonsai trees for handwritten/printed text discrimination," in *Document Recognition and Retrieval Conference*, 2014.

[9] A. Graves and J. Schmidhuber, "Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks," in *Conference on Neural Information Processing Systems*, 2008.

[10] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *Workshop on Automatic Speech Recognition and Understanding*, 2011.

[12] M. Mohri, "Finite-State Transducers in Language and Speech Processing," *Computational Linguistics*, vol. 23, pp. 269–311, 1997.

[13] I. Witten and T. Bell, " The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression," *IEEE Transactions on Information Theory*, vol. 37, no. 4, 1991.