2017 14th IAPR International Conference on Document Analysis and Recognition

Full-Page Text Recognition: Learning Where to Start and When to Stop

Bastien Moysset*§, Christopher Kermorvant†, Christian Wolf‡§
*A2iA SA, Paris, France
†Teklia SAS, Paris, France
‡Université de Lyon, CNRS, France
§INSA-Lyon, LIRIS, UMR5205, F-69621

***Abstract*—Text line detection and localization is a crucial step for full page document analysis, but still suffers from heterogeneity of real life documents. In this paper, we present a new approach for full page text recognition. Localization of the text lines is based on regressions with Fully Convolutional Neural Networks and Multidimensional Long Short-Term Memory as contextual layers.**

In order to increase the efficiency of this localization method, only the position of the left side of the text lines are predicted. The text recognizer is then in charge of predicting the end of the text to recognize. This method has shown good results for full page text recognition on the highly heterogeneous Maurdor dataset.

I. Introduction

Most applications in document analysis require text recognition at page level, where only the raw image is available and no preliminary hand-made annotation can be used. Traditionally, this problem has mainly been addressed by separating the process into two distinct steps; namely the text line detection task, which is frequently proceeded by additional paragraph and word detection steps, and the text recognition task. In this work we propose a method, which couples these two steps tighter by unloading some of the burden from the difficult localization step to the recognition task. In particular, the localization step detects the starts of the text lines only. The problem of finding where to stop the recognition is solved by the recognizer itself.

A. Related work

Numerous algorithms have been proposed for the text line localization. Some are used in a bottom-up approach by grouping sub-components like connected components or black pixels into lines. RLSA [31] uses morphological opening on black pixels to merge the components that belong to the same text line. Similarly, Shi et al. [27] resort to horizontal ellipsoidal steerable filters to blur the image and merge the components of the text line. In [30], gradients are accumulated and filtered. Louloudis et al. [14] employ a Hough algorithm on the connected component centers while Ryu et al. [26] cluster parts of the connected components according to heuristic based successive splits and merges.

Other methods follow a top-down approach and split the pages into smaller parts. The XY-cut algorithm [20] looks for vertical and horizontal white spaces to successively split the pages in paragraphs, lines and words. Similarly, projection profile algorithms like Ouwayed et al. [22] are aimed at finding the horizontal whiter parts of a paragraph. This technique is extended to non-horizontal texts by methods like Nicolaou et al. [21] that dynamically finds a path between the text lines or by Tseng et al. [29] that use a Viterbi algorithm to minimize this path.

Techniques like the ones proposed by Mehri et al. [15] or Chen et al. [5] classify pixels into text or non-text but need post-processing techniques to constitute text lines.

These techniques usually work well on the homogeneous datasets they have been tuned for but need heavy engineering to perform well on heterogeneous datasets like the Maurdor dataset [4]. For this reason, Machine learning has proven to be efficient, in particular deep convolutional networks. Early work from Delakis et al. [6] classifies scene text image parts as text and non-text with a Convolutional Neural Network on a sliding window. In [17], paragraph images are split vertically using a recurrent neural network and CTC alignment. More recently, methods inspired from image object detection techniques like MultiBox [7], YOLO [25] or Single-Shot Detector (SSD) [13] have arisen. Moysset et al. [16] proposed a MultiBox based approach for direct text line bounding boxes detection. Similarly, Gupta et al. [10] and Liao et al. [12] use respectively YOLO based and SSD based approach for scene text detection. Moysset et al. [18] also propose the separate detection of bottom-left and top-right corners of line bounding boxes.

The text recognition part is usually made with variations of Hidden Markov Models [1] or 2D Long Short-Term Memory (2D-LSTM) [8] neural networks.

Finally, Bluche et al. [2] use a hard attention mechanism to directly perform full page text recognition without prior localization. The iterative algorithm finds the next attention point based on the sequence of seen glimpses modeled through the hidden state of a recurrent network.

B. Method overview

In this work, we address full page text recognition in two steps. First, a neural network detects where to start to recognize a text line, and a second network performs the text recognition and decides when to stop the process. More precisely, the former network detects the left sides of each text lines by predicting the value of the object position coordinates

2379-2140/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDAR.2017.147

TABLE I: Network architecture/hyper-parameters. The input and feature map sizes are an illustrative example. The number of parametres does *NOT* depend on the size of the input image.

Layer	Filter size	Stride	Size of the feature maps	Number of parameters
Input	/	/	1×(598×838)	
C1	4×4	3×3	12×(199×279)	204
LSTM1	/	/	" "	8880
C2	4×3	3×2	16×(66×139)	2320
LSTM2	/	/	" "	15680
C3	6×3	4×2	24×(16×69)	6936
LSTM3	/	/	" "	35040
C4	4×3	3×2	30×(5×34)	8670
LSTM4	/	/	" "	54600
C5	3×2	2×1	36×(2×33)	6516
Output	1×1	1×1	4×20×(2×33)	2960

as a regression problem. This detection neural network system is detailed in Part II and the left-side strategy is explained in Part III-A. The latter network recognizes the text and predicts the end of the text of the line as described in Part III-B. The experimental setup is described in Part IV and results are shown and analyzed in Part V.

II. Object localization with deep networks

A. Network description

In the lines of [7], we employ a neural network as a regressor to predict the positions of objects in images. The network predicts a given number N of object candidates. Each of these object candidates is indexed by a linear index n and defined by K coordinates $l_n{=}\{l_n^k\},\ k{=}1\dots K$ corresponding to the position of the object bounding box in the document and a confidence score c_n. As the number of objects in an image is variable, at test time, only the objects with a confidence score over a threshold are kept.

In order to cope with the small amount of training data available for document analysis tasks and to detect a large number of objects corresponding to our text lines, we adopted the method described in [16]. We do not use a fully connected layer at the end of the network that has as inputs features conveying information about the whole page image and, as outputs, all the object candidates of the page. Instead, our method is fully convolutional, which allows the network to share parameters over the different regressors. More precisely, we use a $1{\times}1$ convolution to predict the objects locally and, consequently, to highly reduce the number of parameters in the network.

Layers constituted of Two-Dimensional Long-Short-Term-Memory cells (2D-LSTM) [8] are interleaved between the convolutional layers in order to recover the context information lost by the local nature of the detection.

The architecture is similar to the one in [16]. It is described in Table I and illustrated in Figure 1.

B. Training

We used the same training process as the one described in [7]. The cost function is a weighted sum between a confidence cost and the Euclidean distance between the two object positions (predicted and ground-truth):

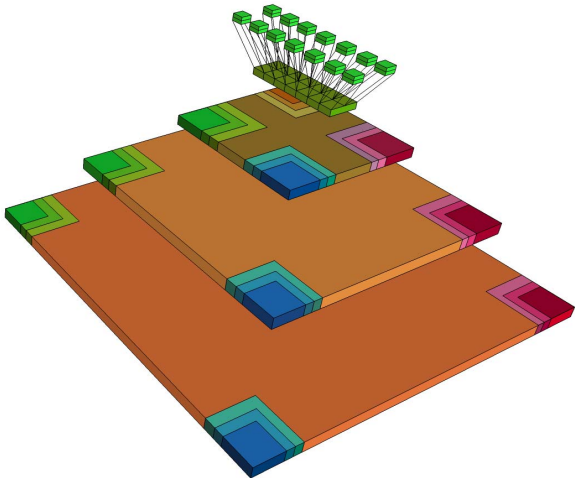


Fig. 1: Sketch of the Convolutional Recurrent Neural Network that locally predicts the object positions (we do not show the correct numbers of layers and units).

$$Cost = \sum_{n=0}^{N}\sum_{m=0}^{M} X_{nm}\left(\alpha\left\|l_n - t_m\right\|^2 - \log(c_n)\right) - \sum_{n=0}^{N}(1 - \sum_{m=0}^{M} X_{nm})\log(1-c_n) \quad (1)$$

Here, the N object candidates have position coordinates l_n and confidence c_n while the M reference objects have position coordinates t_m. α is a parameter weighting localisation and confidence costs. As the output of the network (as well as the ground-truth information) is structured, a matching between the two of them is necessary in order to calculate the loss in equation 1. This matching is modelled through the variable $X{=}\{X_{nm}\}$, a binary matrix. In particular, $X_{nm}{=}1$ if network output n has been matched to ground truth object m in the given image. Equation 1 needs to be minimized under constraints enforcing one-to-one matches, which can be solved efficiently through the Hungarian algorithm [19].

We could not confirm the claims reported in [7] who apply this matching process for object detection in natural images. In particular, no improvement was found when using anchor positions associated to objects which were mined through k-means clustering. On the other hand, we found it useful to employ different weights α for the two different uses of equation 1. A higher value for α was used during matching (solving for X) than for backpropagation (learning of network parameters). This favours the use of all outputs during training — details are given in section V.

III. Localization and recognition

A. Line detection with left-side triplets

The first step detects the left-side of each lines through the network described in Section II. The model predicts

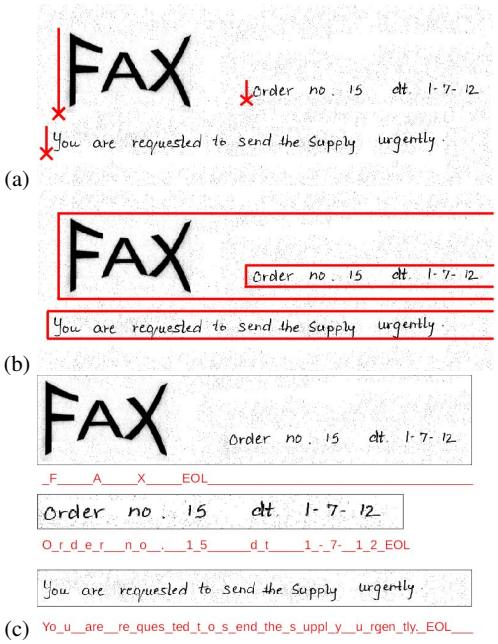


Fig. 2: Description of the triplet-based localization techniques. (a) Detection of left-side triplet objects. (b) Extraction of corresponding text lines. (c) Recognition of text lines with End-of-line characters (EOL).

three position values, i.e. K=3: 2 coordinates for the lower left corner plus the text height. Additionally, a prediction confidence score is output.

We also compare this method with two competing strategies:

i) point localization [18], K=2, where only the x and y coordinates of lower left points are detected,

ii) and full box localization [16], where K=4 and the x and y coordinates of the bottom-left corners of the text line bounding boxes are predicted with the width and the height of the text lines.

We also found that expanding the text box by a 10 pixel margin improves the full-page text recognition rates.

B. End-of-line detection integrated with recognition

Detecting only the left side of the text lines and extending it toward the right part of the image as illustrated in Figure 2 b) means that for documents with complex layouts, some text from other text lines can be present in the image to be recognized.

For this reason, a 2D-LSTM based recognizer similar to the one described in [23] is trained with the Connectionist Temporal Classification [9] (CTC) alignment procedure to recognize the text present in, and only in, the designed text line. We found that the results were slightly improved by adding a End-of-line (EOL) label at the end of the text labels. This means that the network will learn, through the CTC training, to align the text labels with the frames corresponding to these image characters, as usual. But it will also learn to predict when the line is over, mark it with the EOL label, and learn not to predict anything else on the right side of the image. The context conveyed by the LSTM recurrent layers is responsible for this learning ability.

Two different recognition networks are trained, respectively for French and English. They are trained to recognized both printed and handwritten simultaneously.

IV. Experimental Setup

A. Datasets

We evaluate our method on the Maurdor dataset [4], which is composed of 8773 heterogeneous documents in French, English or Arabic, both printed and handwritten (Train: 6592, Validation: 1110, Evaluation: 1071). Because the annotation is given at paragraph level, we used the technique described in [3] to check the quality of line candidates with a constrained text recognition in order to obtain annotation at line level. All these lines are used to train the text recognizers described in section III-B and, on the test set, for the recognition experiments in Table IV.

For training the text line detection systems, only the 5308 pages for which we are confident enough that all the lines are detected are kept (Train: 3995, Validation: 697, Evaluation: 616). This subset is also used for the precision experiments shown in Tables II and III.

Finally, for the end-to-end evaluation results shown in Table V, we kept all the original documents of the test set in only one language, in order to avoid the language identification problem. We obtain 507 pages in French and 265 pages in English.

B. Metrics

Three metrics were used for evaluation :

1) **F-Measure** metrics is used in Tables II and III in order to measure precision of the detected objects being in the neighbourhood of reference objects. A detected object l is considered as correct if it is the closest hypothesis from the reference object t and if $||l^k - t^k|| < T$ for all $k \in [0, K]$ where K is the number of coordinates per object set to 2 for Table II (points) and to 3 in Table III (triplets) and T is the size of the acceptance zone given as a proportion of the page width.
2) **Word Error Rate** (WER) metrics is the word level Levenshtein distance [11] between recognized and reference sequences of text.
3) **Bag of Word** (BOW) metrics is given at page level as a F-Measure of words recognized or not in the page. As explained in [24], it is a proper metric to compute recognition rate at page level because it does not need any alignment or ordering of the text lines that can be ambiguous for unconstrained documents.

C. Hyper-parameters

We trained with the RmsProp optimizer [28] with an initial learning rate of 10^{-3} and dropout after each convolutional layer. The α parameter is set to α=1000 for matching (solving for X) and to α=100 for gradient computation.

TABLE II: Comparison of the F-Measure scores for the detection of bottom-left points with respect to the acceptance zone size for networks trained to detect points, triplets or boxes. Acceptance zones given as proportion of page width.

Acceptance zone size	0.003	0.01	0.03	0.1
Box network ([16], K=4)	6.8%	45.0%	82.8%	89.9%
Point network ([18], K=2)	10.7%	57.4%	85.7%	91.7%
Triplet network (Ours, K=3)	11.2%	58.4%	87.0%	92.6%

TABLE III: Comparison of the F-Measure scores for the detection of left-side triplets with respect to the acceptance zone size for networks trained to detect triplets or boxes. Acceptance zones given as proportion of page width.

Acceptance zone size	0.003	0.01	0.03	0.1
Box network ([16], K=4)	3.4%	24.6%	71.4%	89.7%
Triplet network (Ours, K=3)	4.2%	47.2%	84.8%	92.4%

V. Experimental results

A. Precision of the object localizations

Similarly to what is described in [18], we observed some instability in the position of the predicted objects when trying to detect boxes. Our intuition is that precisely detecting objects which ends outside of the convolutional receptive field of the outputs is difficult.

Characters may have a size of 1 or 2 mm in standard printed pages, corresponding to 0.005 and 0.01 as a proportion of the page width. Interlines may have similar sizes. Therefore, it is important that the position prediction is close enough in order not to harm the text recognition process.

The method described in [18] was dealing with this problem by detecting separately the bottom-left and top-right points and posteriorly pairing them. We observed that the precision was not harmed by the detection of triplets of coordinates (left, top, bottom).

In Table II, we show the F-measure of the detection of left-bottom points for several acceptance zone sizes. The results emphasize that detecting full text boxes reduces precision. Meanwhile, the precision of bottom-left point prediction is equivalent when the network is trained to detect triplets and not points.

Table III shows the same experiment with a 3D acceptance zone defined on the triplet positions, showing the same improved results for the triplet detection for small acceptance zones.

B. Detection of the line end with the text recognizer

In Table IV we compared two text line recognizers trained respectively on the reference text line images and on text line images defined only by the left sides coordinates of the text line and extended toward the right end of the page. These two recognizers are evaluated in both cases with the WER metric.

While the network trained on reference boxes is obviously not working well on extended test images, we see that the network trained on extended lines works on both tasks nearly as well as the network trained on reference boxes.This confirms that we can rely on the text recognizer to ignore the part of the line that does not belong to the text line.

TABLE IV: Text recognition Word Error Rates (WER) for networks trained/evaluated on reference boxes or box defined only by their left sides.

	Evaluated on reference boxes	Evaluated with left-sides only
Trained on reference boxes	9.0%	46.7%
Trained with left-sides only	10.6%	9.8%

TABLE V: Comparison of full-page recognition systems with the Bag of Words (BOW) metric on the French and English documents of the Maurdor dataset.

System	French dataset	English dataset
Shi et al. [27]	48.6%	30.4%
Nicolaou et al. [21]	65.3 %	50.0 %
Erhan et al.	65.3 %	50.0 %
Multibox [7]	27.2%	14.8%
Multibox [7] (optimized)	32.4%	36.2%
Box network [16]	71.2%	71.1%
Points network [18]	71.7%	72.3%
Triplet network (proposed)	79.9%	79.1%

C. Full page text recognition

Finally, we compared our method with baselines and concurrent approaches for full page recognition. The evaluation was carried out using the BOW metric and is shown on Table V. We show that the proposed methods yield good results on both the French and English subsets, consistently overpassing the document analysis baselines based on image processing, the object localisation baseline and the concurrent box detection and paired point detection systems. At decoding time, for an image of size 598x838, the object detection part takes a mean time of 245 ms per page, which is faster than the text recognition part, done in higher resolution, that takes 638 ms per page on average. Performances are CPU only on Intel Xeon E5-2640-v4 with 64 MB of RAM.

Some illustrations of the left-side triplets detection alongside with the final full-page text recognition are given in Figure 3 and emphasize the ability of the system to give good results on various types of documents.

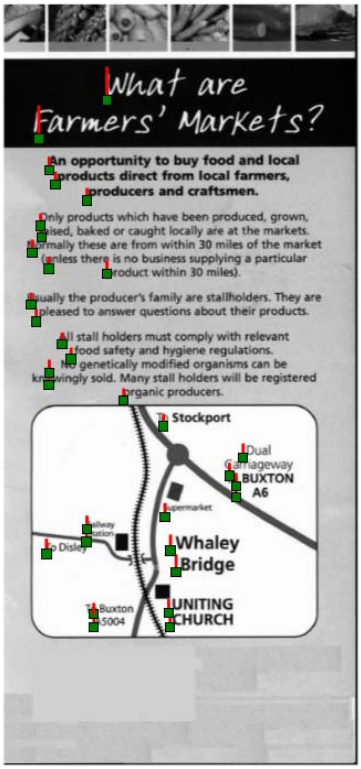
VI. Conclusion

We described a full page recognition system for heterogeneous unconstrained documents that is able to detect and recognize text in different languages. The use of a neural network localisation process helps to be robust to the intra-dataset variations. In order to simplify the process and to gain both in precision and in preciseness, we focus on predicting the starting point (left) of the text line bounding boxes and leave the prediction of the end point (right) to a 2D-LSTM based text recognizer. We report excellent results on the Maurdor dataset and show that our method outperform both image-based and concurrent learning-based methods.

Fig. 3: Illustration of full-page recognition results.

(a) Input+Localization results

(b) Input+Localization results



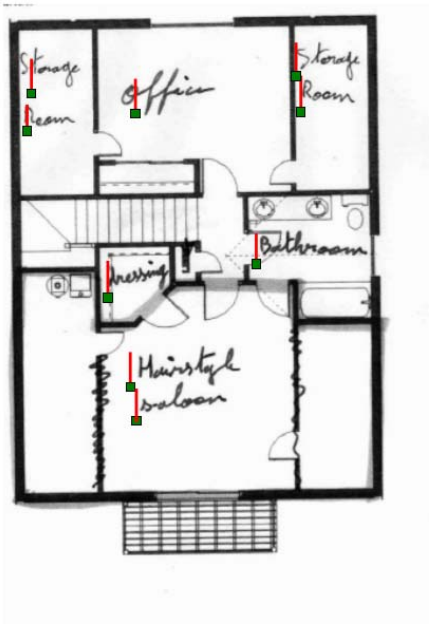
(c) Recognized text

- Invitation pour mon
- la Famille Rochefleur :
- Nous avons le plaisir de vous inviter à l'ouverture
- exceptionnelle du salon du mariage qui se tiendra à chargey
- le verte-end du 6 et 7 Octobre 2007 de 20h à 19h.
- Vous aurez ainsi le plaisir d'assister à un
- défilé de robes de mariées, toutes plus oublimes les unes.
- que les autres ; de rencontrer des animateurs de
- sinées, des décoracteurs, des joailliers, des traiteuns et :
- bien plus encore !
- Grâce à cette invitation vous pourrez bénéficier de :
- de prenmations exceptionnelles et vous pourrez
- également profiter d'un délicieux buffet, mis à disposition!
- dans le sait bout d'égaler vos papilles!
- Une Tombola sera également organisée vous donnant.
- ainsi l'opportunité de gagner de nombreux cadeaux.
- En plus d'un séjour pour deux personnes, tout frais
- passés à l'Ile haunité.
- En espérant votre présence au salon, je vous
- demanderai de remplir le talon réponse ci-joint dans
- les meilleurs délais !
- Cordialement,
- l'organisatrice de
- LEU******
- PREVOST Myriam
- *******

(d) Recognized text

- DY
- ****4481991991
- An opportunity to buy food and local
- products direct from local farmers,
- 1cts direct from local farmers,
- Only products which have been produced, grown,
- raised, baked or caught locally are at the markets.
- formally these are from within 30 miles of the market
- (unless there is no business supplying a particular
- product within 30 miles).
- Usually the producer's family are stallholders. They are
- pleased to answer questions about their products.
- All staall holders must comply with relevant
- food safety and hygiene regulations.
- No genetically modified organisms can be
- knowingly sold. Many staall holders will be registered
- organic producers.
- To Stockport
- Dual
- Carriageway
- to BUXTON
- A6
- supermarket
- Mailway
-
- To Disien-
- Whalay
- Bridge
- To Burton
- A5004
- UNITTING
- CHURCH

(e) Input+Localization results



(f) Recognized text

- tonage
- Room
- offi***
- Storage
- Room
- Faithroom
- Merrill
- Mainstake
- Saloon

REFERENCES

[1] Bertolami, R., Bunke, H.: Hidden markov model-based ensemble methods for offline handwritten text line recognition. Pattern Recognition 41(11), 3452–3460 (2008)

[2] Bluche, T.: Joint line segmentation and transcription for end-to-end handwritten paragraph recognition. In: Advances in Neural Information Processing System (2016)

[3] Bluche, T., Moysset, B., Kermorvant, C.: Automatic line segmentation and ground-truth alignment of handwritten documents. In: Int. Conf. on Frontiers in Handwriting Recognition (2014)

[4] Brunessaux, S., Giroux, P., Grilheres, B., Manta, M., Bodin, M., Choukri, K., Galibert, O., Kahn, J.: The maurdor project - improving automatic processing of digital documents (2014)

[5] Chen, K., Seuret, M., Liwicki, M., Hennebert, J., Ingold, R.: Page segmentation of historical document images with convolutional autoencoders. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 1011–1015. IEEE (2015)

[6] Delakis, M., Garcia, C.: text detection with convolutional neural networks. In: VISAPP (2). pp. 290–294 (2008)

[7] Erhan, D., Szegedy, C., Toshev, A., Anguelov, D.: Scalable object detection using deep neural networks. In: IEEE Conf. on Computer Vision and Pattern Recognition (2014)

[8] Graves, A., Schmidhuber, J.: Offline handwriting recognition with multidimensional recurrent neural networks. In: NIPS (2009)

[9] Graves, A., Fernandez, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In: International Conference on Machine Learning. pp. 369–376 (2006)

[10] Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2315–2324 (2016)

[11] Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. In: Soviet physics doklady. vol. 10, pp. 707–710 (1966)

[12] Liao, M., Shi, B., Bai, X., Wang, X., Liu, W.: Textboxes: A fast text detector with a single deep neural network. In: AAAI (2017)

[13] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.: Ssd: Single shot multibox detector. In: European Conf. on Computer Vision (2016)

[14] Louloudis, G., Gatos, B., Pratikakis, I., Halatsis, C.: Text line and word segmentation of handwritten documents. Pattern Recognition 42(12), 3169–3183 (Dec 2009)

[15] Mehri, M., Héroux, P., Gomez-Krämer, P., Boucher, A., Mullot, R.: A pixel labeling approach for historical digitized books. In: Document Analysis and Recognition (ICDAR), 2013 12th International Conference on. pp. 817–821. IEEE (2013)

[16] Moysset, B., Kermorvant, C., Wolf, C.: Learning to detect and localize many objects from few examples. arXiv preprint arXiv:1611.05664 (2016)

[17] Moysset, B., Kermorvant, C., Wolf, C., Louradour, J.: Paragraph text segmentation into lines with recurrent neural networks. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on. pp. 456–460. IEEE (2015)

[18] Moysset, B., Louradour, J., Kermorvant, C., Wolf, C.: Learning text-line localization with shared and local regression neural networks. In: Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on. pp. 1–6. IEEE (2016)

[19] Munkres, J.: Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics 5(1), 32–38 (1957)

[20] Nagy, G., Seth, S.: Hierarchical representation of optically scanned documents. In: Proceedings of International Conference on Pattern Recognition. vol. 1, pp. 347–349 (1984)

[21] Nicolaou, A., Gatos, B.: Handwritten Text Line Segmentation by Shredding Text into its Lines. International Conference on Document Analysis and Recognition (2009)

[22] Ouwayed, N., Belaïd, A.: A general approach for multi-oriented text line extraction of handwritten documents. International Journal on Document Analysis and Recognition (IJDAR) 15(4), 297–314 (2012)

[23] Pham, V., Bluche, T., Kermorvant, C., Louradour, J.: Dropout improves recurrent neural networks for handwriting recognition. In: Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on. pp. 285–290. IEEE (2014)

[24] Pletschacher, S., Clausner, C., Antonacopoulos, A.: Europeana newspapers ocr workflow evaluation. In: Workshop on Historical Document Imaging and Processing (2015)

[25] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: IEEE Conf. on Computer Vision and Pattern Recognition (June 2016)

[26] Ryu, J., Koo, H.I., Cho, N.I.: Language-independent text-line extraction algorithm for handwritten documents. IEEE Signal processing letters 21(9), 1115–1119 (2014)

[27] Shi, Z., Setlur, S., Govindaraju, V.: A Steerable Directional Local Profile Technique for Extraction of Handwritten Arabic Text Lines. In: International Conference on Document Analysis and Recognition (2009)

[28] Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning 4(2) (2012)

[29] Tseng, Y.H., Lee, H.J.: Recognition-based handwritten chinese character segmentation using a probabilistic viterbi algorithm. Pattern Recognition Letters 20(8), 791–806 (1999)

[30] Wolf, C., Jolion, J.M., Chassaing, F.: Text Localization, Enhancement and Binarization in Multimedia Documents. In: ICPR. vol. 2, pp. 1037–1040 (2002)

[31] Wong, K.Y., Casey, R.G., Wahl, F.M.: Document analysis system. IBM journal of research and development 26(6), 647–656 (1982)