To appear on International Conference on Document Analysis and Recognition (ICDAR 2013), Washington D.C., 2013

On Combining Multiple Segmentations in Scene Text Recognition

Lukáš Neumann
Centre for Machine Perception, Department of Cybernetics
Czech Technical University, Prague, Czech Republic
neumalu1@cmp.felk.cvut.cz

Jiří Matas
Centre for Machine Perception, Department of Cybernetics
Czech Technical University, Prague, Czech Republic
matas@cmp.felk.cvut.cz

***Abstract*—An end-to-end real-time scene text localization and recognition method is presented. The three main novel features are: (i) keeping multiple segmentations of each character until the very last stage of the processing when the context of each character in a text line is known, (ii) an efficient algorithm for selection of character segmentations minimizing a global criterion, and (iii) showing that, despite using theoretically scale-invariant methods, operating on a coarse Gaussian scale space pyramid yields improved results as many typographical artifacts are eliminated.**

The method runs in real time and achieves state-of-the-art text localization results on the ICDAR 2011 Robust Reading dataset. Results are also reported for end-to-end text recognition on the ICDAR 2011 dataset.

I. Introduction

Text localization and recognition in real-world images (i.e. *scene text* localization and detection, photo OCR) is a field of computer vision which has recently received significant attention. Several competitions have been held in the past years [7], [8], [14]. The winning method in the most recent one achieved only localization recall of 62% [14], which makes automatic text recognition still impractical for applications. On the other hand, when accuracy of such methods is improved to a sufficient level, there are many possible applications - reading out text to visually impaired people, translating language with an automatic input of text written in an unknown script and indexing large image/video databases by their textual content (e.g. Google Street View, TV news archives, etc.).

Localizing text in an image can be a computationally very expensive task as generally any of the 2^N subsets can correspond to text (where N is the number of pixels). The search for text is often limited to only a specific subset of rectangles of an image ("sliding window" methods) where a trained classifier decides whether it contains text or not [1], [4], [6]. The drawback of such an approach is that the number of rectangles that needs to be evaluated grows rapidly when text with different scale, aspect, rotation and other distortions has to be found.

Another and recently more popular approach, which is also exploited by our method, is to first detect individual characters using local properties (color, stroke-width, etc.) assuming that the selected property does not change much for neighboring characters (region-based methods) and then group the characters into higher structures such as words and text lines in subsequent stages [2], [10]–[13], [18], [20]. The advantage is that the complexity typically does not depend on the parameters of the text (range of scales, orientations, fonts) and that the methods typically provide a character segmentation which can be used in the OCR stage.

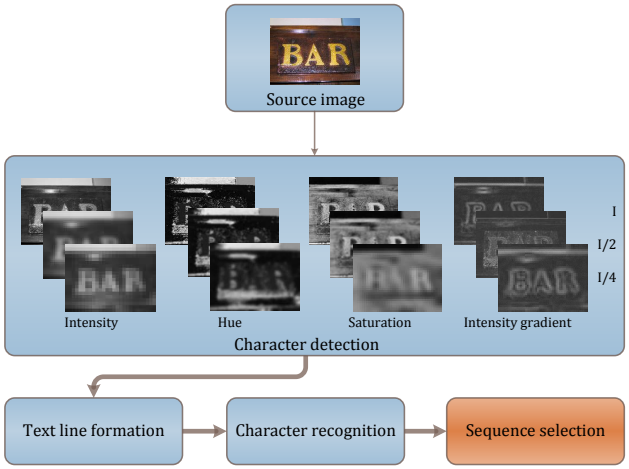

Fig. 1. Overview of the method. Multiple segmentations from the text detector are kept until very last stage where the final sequence of characters is selected taking into account the context of the text line

In this paper, we show that it is beneficial to keep multiple segmentations of each character until the very last stage of the processing when context of each character in a text line is known and we propose an efficient algorithm to select the final sequence of characters. We also show that although region-based methods are scale invariant up to discretisation effects, a pre-processing with a Gaussian pyramid yields improved recall because many typographical artifacts are eliminated (e.g. joint letters, characters consisting of small regions). The impact of novel contributions presented in this paper is also demonstrated as a significant improvement of the overall performance over the previously published NM12 method [11].

For the standard ICDAR 2011 dataset and protocol [14], the proposed method achieves state-of-the-art results in text localization (f-measure 75.4%) and improves the text recognition results[1] previously published in [11]. The processing is near real-time on a standard PC, i.e. the processing time is comparable with the time it would take a human to read the text.

The rest of the document is structured as follows: In

[1]To our knowledge, our method is the only method where end-to-end text recognition results on the standard ICDAR 2011 dataset are published

Section II, an overview of previously published methods is given. Sections III and IV describe the proposed method. In Section V, the experimental evaluation is presented. The paper is concluded in Section VI.

II. Previous Work

Existing methods for general text localization can be categorized into two major groups - methods based on a sliding window and methods based on regions (characters) grouping. Methods in the first category [1], [5] use a window which is moved over the image and the presence of text is estimated on the basis of local image features. While these methods are generally more robust to noise in the image, their computational complexity is high because of the need to search with many rectangles of different sizes, aspect ratios and potentially rotations. Additionally, support for slanted or perspectively distorted text is limited and sliding window methods do not always provide accurate enough text segmentation which can be used for character recognition [1].

The majority of recently published methods for text localization falls into the latter category [2], [9], [18], [20]. The methods differ in their approach to individual character detection, which could be based on edge detection, character energy calculation or extremal region detection. While the methods are paying great attention to individual character detection, the decision about final segmentation is done at very low level using only local features.

Additionally, the methods focus solely on text localization, i.e. they estimate the position of the text, but do not provide its content. Our method (first presented in [12]) was the first one to show end-to-end text localization and recognition. One of few methods that perform both text localization and recognition is the method of Wang et al. [16] which finds individual characters as visual words using the sliding-window approach and then uses a lexicon to group characters into words. The method is able to cope with noisy data, but its generality is limited as a lexicon of words (which contains at most 500 words in their experiments) has to be supplied for each individual image.

Several competitions [7], [8], [14] have been held in this field to evaluate text localization performance of the methods. Unfortunately end-to-end text localization and recognition was not part even of the most recent competition [14] and therefore no comparison with other methods is available.

III. Character Detector

In the proposed method, individual characters are detected as *Extremal Regions*. Let us consider an image $\mathbf{I}$ as a mapping $\mathbf{I} : \mathcal{D} \subset \mathbb{N}^2 \rightarrow \mathcal{V}$, where $\mathcal{V}$ typically is $\{0, \ldots, 255\}^3$ (a color image). A projection image $\mathbf{C}$ of the image $\mathbf{I}$ is a mapping $\mathbf{C} : \mathcal{D} \rightarrow \mathcal{S}$ where $\mathcal{S}$ is a totally ordered set (typically $\{0, \ldots, 255\}$) and $f_c : \mathcal{V} \rightarrow \mathcal{S}$ is a *color space projection* of pixel values to a totally ordered set. Let A denote an *adjacency* (neighborhood) relation $A \subset \mathcal{D} \times \mathcal{D}$. Region $\mathcal{R}$ of an image $\mathbf{I}$ is a contiguous subset of $\mathcal{D}$, i.e. $\forall p_i, p_j \in \mathcal{R}\ \exists p_i, q_1, q_2, \ldots, q_n, p_j : p_i A q_1, q_1 A q_2, \ldots, q_n A p_j$. Outer region boundary $\partial\mathcal{R}$ is a set of pixels adjacent but not belonging to $\mathcal{R}$, i.e. $\partial\mathcal{R} = \{p \in \mathcal{D} \setminus \mathcal{R} : \exists q \in \mathcal{R} : pAq\}$. *Extremal Region* (ER) is a region whose outer boundary pixels have strictly higher values than the region itself, i.e. $\forall p \in \mathcal{R}, q \in \partial\mathcal{R} : \mathbf{C}(q) > \theta \geq \mathbf{C}(p)$, where θ denotes *threshold* of the Extremal Region.

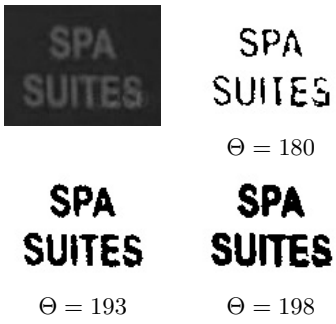

Fig. 2. Character boundaries are often fuzzy and it is not possible to locally determine the threshold value unambiguously. Note the binarization of letters "ITE" in the word "SUITES" - as the threshold Θ is increased their appearance goes from "IIE" through "ITE" to "m"

Assuming that a single character can be detected as an Extremal Region, three parameters still have to be determined: the threshold θ, the adjacency relation A and the color space projection f_c. In the proposed method, parameter values are efficiently enumerated for each region individually and optimal values are selected at a later stage by a trained cost function which exploits context of the character in its text line (see Section IV). This contrasts with all previously published methods ([2], [12], [18]), where final decision about character segmentation is done in an very early stage without any context of a text line.

Let us further discuss why determining a single value of the text detector parameters is not always a straightforward task and how the proposed method approaches this problem:

Threshold. As demonstrated in Figure 2, character boundaries are often fuzzy and it is not possible to locally determine the threshold value unambiguously. In the proposed method, all thresholds are evaluated and the ones which most likely correspond to a valid character segmentation are selected using a CSER classifier [11]. This efficiently reduces the number of thresholds for each character to a few most distinguished segmentations, in complexity linear in number of pixels.

Adjacency. There are many instances in real-world images where characters are formed of smaller elements (see Figure 3a) or a single element consists of multiple joint characters (see Figure 3d). In both cases, a successful detection of the characters as Extremal Regions depends on the definition of the adjacency relation, i.e. which pixels belong to a single component. In the proposed method, a standard 4-neighborhood adjacency relation is adapted, but the image is pre-processed with a Gaussian pyramid. This process alters the adjacency relation so that in each level of the pyramid only a certain interval of character stroke widths is amplified - if a character consists of multiple elements, the elements are merged together into a single region (see Figure 3c). Furthermore, serifs and thin joints between multiple characters are eliminated (see Figure 3e). This process does not represent a major overhead as each level is 4 times faster than the previous one (the image is 4 times smaller).

Color Space Projection. Because Extremal Regions are

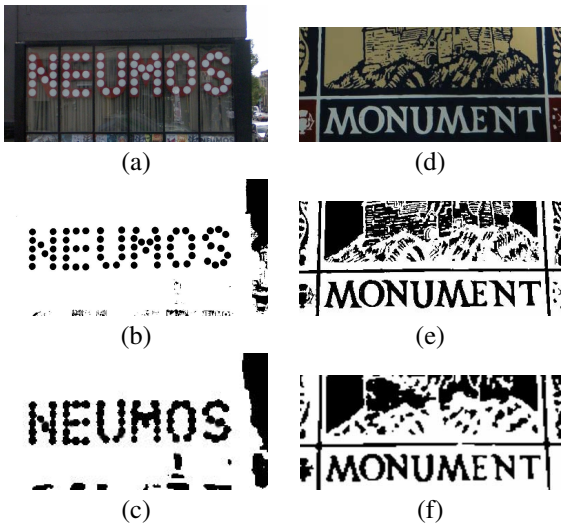

Fig. 3. Processing with a Gaussian pyramid. Characters formed of multiple small regions (a,b) merge together and a single region corresponds to a single character (c). A single region which corresponds to characters "ME" (d) for which there does not exist any threshold in the original image (e) is broken into two and serifs are eliminated (f)

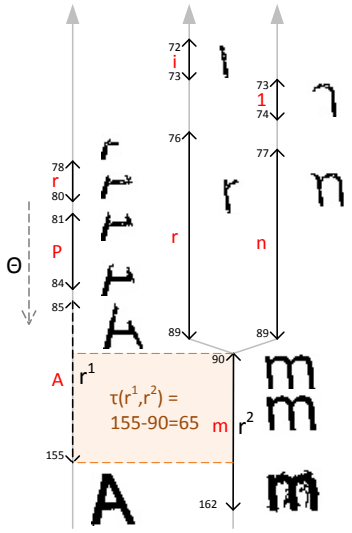

Fig. 4. Threshold interval overlap $\tau(r^1, r^2)$. A threshold interval is an interval of thresholds during which the region has not changed its OCR label (red). Note that as the threshold is increased the region grows or merges with other regions and the label changes

defined on a totally ordered set, it necessary to transform each color image into a single-channel image through a color space projection. There is an infinite number of such projections and a character may or may not be detected as an Extremal Region depending on the selected projection. Therefore, the selection of projections is a trade-off between a higher number of detected characters and faster running time. As previously shown in [11], the combination of intensity, intensity gradient, hue and saturation projections yields the best results.

IV. Sequence Selection as an Optimal Path Problem

Let us consider a *word* as a sequence of characters. Given a set of regions $\mathcal{R}$ from the character detector, it is necessary to select region sequences where in each sequence each region corresponds to a character and the order of the regions in the sequence corresponds to the order of the characters in the word. Note that a solution might not be unambiguous because typically there will be multiple regions (segmentations) that correspond to a single character.

In the proposed method, regions are first agglomerated into *text lines* by efficiently pruned exhaustive search [10], which estimates the text direction on each triplet of regions and then constraints induced by the text direction contribute to the similarity measure used for clustering. A *text line* is a set which consists of regions which share the same text direction and typically will contain regions which correspond to characters (possibly several different segmentation of the same character) as well as regions which represent only a part of a character or clutter regions which just happen to be on the text line.

In the next stage, each region r in the text line is labeled with one or more codes $l(r) = \{c_1 \dots c_n\}, c_i \in \mathcal{A}$ by the character recognition module [12], which was trained on synthetic fonts. The alphabet $\mathcal{A}$ are characters in a Unicode representation. Regions with low confidence are rejected, which eliminates clutter regions that were included in the text line formation stage.

Given regions r_1 and r_2 in a text line, r_1 is a *predecessor* of r_2 $(r_1 \mathcal{P} r_2)$ if r_1 and r_2 are part of the same text line and the character associated with r_1 immediately precedes the one associated with r_2 in the sequence of characters. The predecessor relation $\mathcal{P}$ induces a directed graph G for each text line (see Figure 5), where nodes correspond to labeled regions.

$$\begin{aligned} G &= (V, E) \\ V &= \{r_c \in \mathcal{R} \times \mathcal{A} : c \in l(r)\} \\ E &= \{(r^1_{c_1}, r^2_{c_2}) : r^1 \mathcal{P} r^2\} \end{aligned} \tag{1}$$

where r_c denotes a region r with a label $c \in \mathcal{A}$.

In the proposed method, the relation $\mathcal{P}$ is approximated by a heuristic function $\hat{\mathcal{P}}$ that selects the nearest neighboring region in the left-to-right direction, given that several binary constraints are satisfied (height ratio, area ratio and scale-normalized horizontal distance).

Each node r_c and edge $(r^1_{c_1}, r^2_{c_2})$ has an associated score $s(r_c)$ and $s(r^1_{c_1}, r^2_{c_2})$ respectively

$$s(r_c) = \alpha_1 \psi(r) + \alpha_2 \omega(r_c) \tag{2}$$

$$s(r^1_{c_1}, r^2_{c_2}) = \alpha_3 \tau(r^1, r^2) + \alpha_4 \lambda(c_1, c_2) \tag{3}$$

where $\alpha_1 \dots \alpha_4$ denote weights which are determined in a training stage.

Region text line positioning $\psi(r)$ is calculated as a negative sum of squared Euclidian distances of the region's top and bottom points from estimated position of top and bottom text line respectively. This unary term is incorporated to prefer regions which better fit on the text line.

Character recognition confidence $\omega(r_c)$ estimates the probability, that the region r has the character label c. The estimate is supplied by the character recognition module [12].

Threshold interval overlap is a binary term which is incorporated to express preference for characters having similar threshold interval. A *threshold interval* is an interval of thresholds during which the region has not changed its OCR

TABLE I. COMPARISON OF SELECTING A SINGLE SEGMENTATION AT AN EARLY STAGE AGAINST COMBINING MULTIPLE SEGMENTATIONS AND THE IMPACT OF PREPROCESSING THROUGH WITH A GAUSSIAN PYRAMID

method	recall	precision	f	avg. time per image (s)
SB+SS	45.9	69.8	55.4	1.87
SB+MS	55.5	75.2	63.8	2.35
SWT+SS	38.0	66.0	48.0	**0.60**
SWT+MS	41.0	80.0	54.0	0.84
MB+SS	62.1	85.9	72.0	2.52
MB+MS	**67.5**	**85.4**	**75.4**	3.10

label. A *threshold interval overlap* $\tau(r^1, r^2)$ is the intersection of intervals of regions r^1 and r^2 (see Figure 4).

Transition probability $\lambda(c_1, c_2)$ estimates the probability that the character c_1 follows after the character c_2. Transition probabilities are calculated in a training phase from a dictionary for a given language.

As a final step of the method, the directed graph is constructed with corresponding scores assigned to each node and edge (see Figure 5), the scores are normalized by width of the area that they represent (i.e. node scores are normalized by the width of the region and edge scores are normalized by the width of the gap between regions) and a standard dynamic programming algorithm is used to select the path with the highest score. The sequence of regions and their labels induced by the optimal path is the output of the method (*a word* or a sequence of *words*).

V. EXPERIMENTS

The ICDAR 2011 Robust Reading competition dataset [14] contains 1189 words and 6393 letters in 255 images.

In the first experiment, the impact of selecting a single segmentation in the text detection stage was evaluated by replacing the text detection block of the pipeline (see Figure 1) with another text detection component which outputs only a single segmentation per character.

In the first configuration (SB), the text detector always selected *single best* segmentation for each region. This was achieved by selecting the threshold with highest quality [11], which can be viewed as selecting the segmentation which has the highest probability being a character. In the second configuration (SWT), the Stroke Width Transform [2] was used as character detector. The SWT detector also relies on local decisions (edge detector, stroke width consistency heuristics) to select a single segmentation. In this configuration, the rest of the pipeline remained the same, but the sequence selection (see Section IV) had no effect because there was always just a single possible path through the graph.

TABLE II. COMPARISON WITH MOST RECENT RESULTS ON THE ICDAR 2011 DATASET.

method	recall	precision	f	publication year
proposed method (NM13)	**67.5**	**85.4**	**75.4**	-
Shi's method [15]	63.1	83.3	71.8	2012
Kim's method [14]	62.5	83.0	71.3	not published
NM12 [11]	64.7	73.1	68.7	2012
Yi's Method [19]	58.1	67.2	62.3	2011
TH-TextLoc System [3]	57.7	67.0	62.0	2009

These configurations were compared with the proposed method (*multiple best* - MB). Each text detector was also evaluated in single-scale (SS) and multi-scale (MS) configuration in order to show the impact of preprocessing with a Gaussian pyramid (see Section III).

The results (see Table I) clearly show that selecting the final segmentation at a later stage (MB) yields better localization results than making the final decision in the text detector (SB, SWT). The multi-scale version (MS) always achieves better localization results than the single-scale one (SS), for an overhead of about additional 25% running time. All experiment were run on a single core on a standard PC, so a speed up through multi-thread implementation is an obvious option.

In the second experiment, the proposed method was compared with the most recently published methods (see Table II). Using the ICDAR 2011 competition evaluation scheme [17], the method achieves recall 67.6%, precision 81.1% and f-measure 75.4% in text localization (see Figure 6 for sample outputs). This represents a significant 4 percentage point improvement over the best published result (the ICDAR 2011 Robust Reading competition winner [14]) and a 7 percentage point improvement over the NM12 method [11], which demonstrates the impact of novel contributions presented in this paper. In text recognition, the method achieves recall of 37.8% and precision of 39.4% (calculated using ICDAR 2003 protocol [7], where word is considered a match if all its characters match with the ground truth), which represents a 1

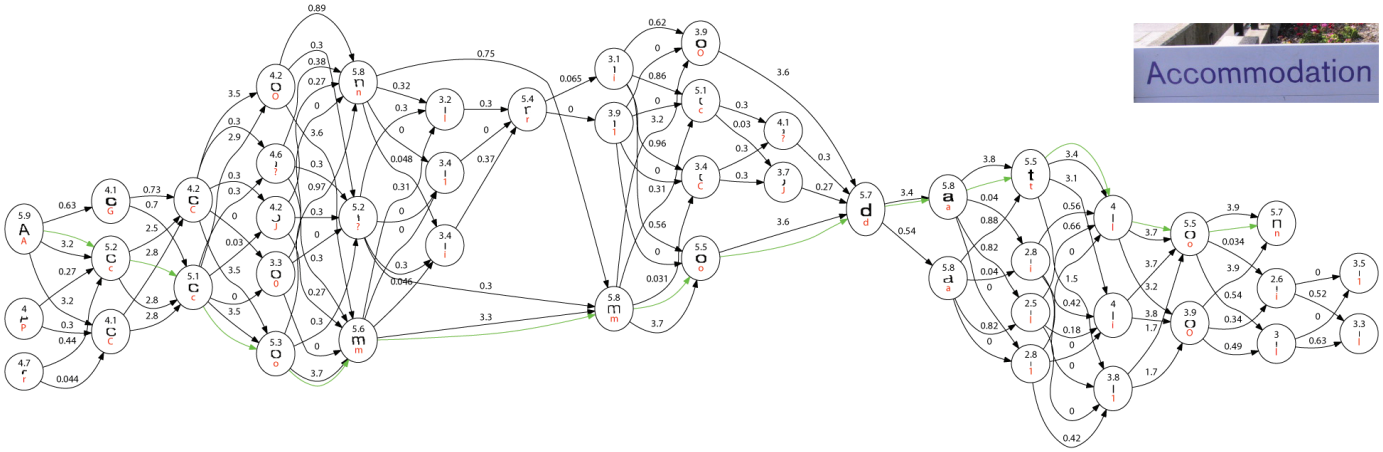

Fig. 5. The final segmentation sequence is found as an optimal path in a directed graph where nodes correspond to labeled regions and edges represent a "is-a-predecessor" relation. Each node r_c represents a region (segmentation) r with a label c (red). Node (edge) score $s(r_c)$ resp. $s(r^1_{c_1}, r^2_{c_2})$ denoted above the node resp. edge. Optimal path denoted in green

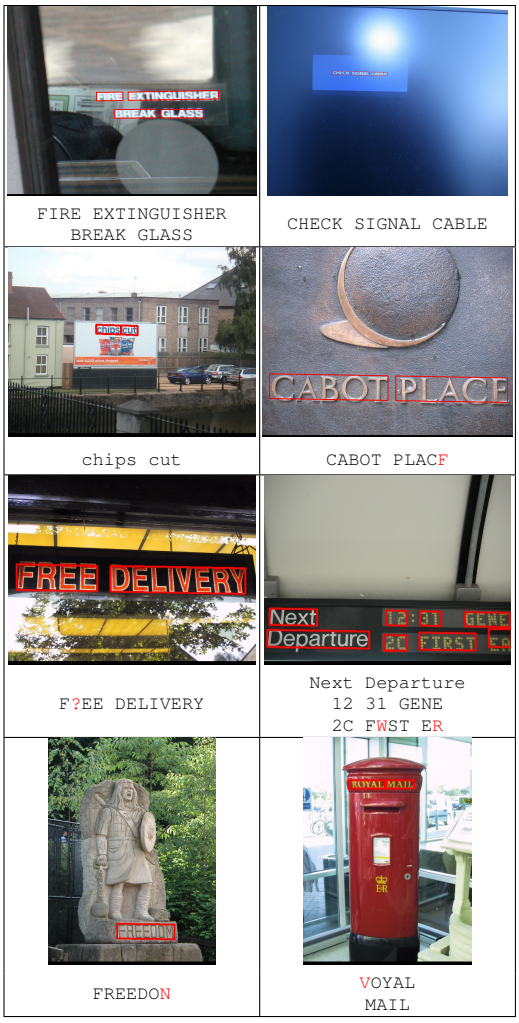

Fig. 6. Text localization and recognition examples on the ICDAR 2011 dataset

percentage point improvement over the previous method.

VI. Conclusion

We have demonstrated that combining segmentations from multiple detection channels and keeping multiple segmentations until last stage of the processing significantly improves the overall performance text localization and recognition results. We have also shown that although region-based methods are themselves scale-invariant, a pre-processing with a Gaussian pyramid yields improved recall because many typographical artifacts are eliminated (e.g. joint letters, characters consisting of small regions). On the standard ICDAR 2011 dataset [14], the method achieves state-of-the-art results in text localization (f-measure 75.4%) and improves previously published text localization results.

Future work includes dealing with current limitations of the method: an inability to detect single- or two-letter words if they are not part of a longer text line, an assumption of a straight base-line and an inability to exclude clutter regions at a beginning or an end of a text line in the sequence selection stage.

Acknowledgment

The authors were supported by the Czech Science Foundation Project GACR P103/12/G084. The authors would also like to acknowledge the Google Research Award.

References

[1] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," *CVPR*, vol. 2, pp. 366–373, 2004.

[2] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *CVPR 2010*, pp. 2963 –2970.

[3] S. M. Hanif and L. Prevost, "Text detection and localization in complex scene images using constrained adaboost algorithm," in *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*. IEEE, 2009, pp. 1–5.

[4] L. Jung-Jin, P.-H. Lee, S.-W. Lee, A. Yuille, and C. Koch, "Adaboost for text detection in natural scene," in *ICDAR 2011*, 2011, pp. 429–434.

[5] J.-J. Lee, P.-H. Lee, S.-W. Lee, A. Yuille, and C. Koch, "Adaboost for text detection in natural scene," in *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, sept. 2011, pp. 429 –434.

[6] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images and videos," *Circuits and Systems for Video Technology*, vol. 12, no. 4, pp. 256 –268, 2002.

[7] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 robust reading competitions," in *ICDAR 2003*, 2003, p. 682.

[8] S. M. Lucas, "Text locating competition results," *Document Analysis and Recognition, International Conference on*, vol. 0, pp. 80–85, 2005.

[9] A. Mishra, K. Alahari, and C. V. Jawahar, "Top-down and bottom-up cues for scene text recognition," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, june 2012, pp. 2687 –2694.

[10] L. Neumann and J. Matas, "Text localization in real-world images using efficiently pruned exhaustive search," in *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, sept. 2011, pp. 687 –691.

[11] ——, "Real-time scene text localization and recognition," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 6 2012, pp. 3538 –3545.

[12] ——, "A method for text localization and recognition in real-world images," in *ACCV 2010*, ser. LNCS 6495, vol. IV, November 2010, pp. 2067–2078.

[13] Y.-F. Pan, X. Hou, and C.-L. Liu, "Text localization in natural scene images based on conditional random field," in *ICDAR 2009*. IEEE Computer Society, 2009, pp. 6–10.

[14] A. Shahab, F. Shafait, and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images," in *ICDAR 2011*, 2011, pp. 1491–1496.

[15] C. Shi, C. Wang, B. Xiao, Y. Zhang, and S. Gao, "Scene text detection using graph model built upon maximally stable extremal regions," *Pattern Recognition Letters*, vol. 34, no. 2, pp. 107 – 116, 2013.

[16] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *ICCV 2011*, 2011.

[17] C. Wolf and J.-M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *Int. J. Doc. Anal. Recognit.*, vol. 8, pp. 280–296, August 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1165154.1165159

[18] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, june 2012, pp. 1083 –1090.

[19] C. Yi and Y. Tian, "Text string detection from natural scenes by structure-based partition and grouping," *Image Processing, IEEE Transactions on*, vol. 20, no. 9, pp. 2594 –2605, sept. 2011.

[20] J. Zhang and R. Kasturi, "Character energy and link energy-based text extraction in scene images," in *ACCV 2010*, ser. LNCS 6495, vol. II, November 2010, pp. 832–844.