Object Proposals for Text Extraction in the Wild

Lluís Gómez and Dimosthenis Karatzas
Computer Vision Center, Universitat Autònoma de Barcelona
Email: {lgomez,dimos}@cvc.uab.es

Abstract—**Object Proposals is a recent computer vision technique receiving increasing interest from the research community. Its main objective is to generate a relatively small set of bounding box proposals that are most likely to contain objects of interest. The use of Object Proposals techniques in the scene text understanding field is innovative. Motivated by the success of powerful while expensive techniques to recognize words in a holistic way, Object Proposals techniques emerge as an alternative to the traditional text detectors.**

In this paper we study to what extent the existing generic Object Proposals methods may be useful for scene text understanding. Also, we propose a new Object Proposals algorithm that is specifically designed for text and compare it with other generic methods in the state of the art. Experiments show that our proposal is superior in its ability of producing good quality word proposals in an efficient way. The source code of our method is made publicly available[1].
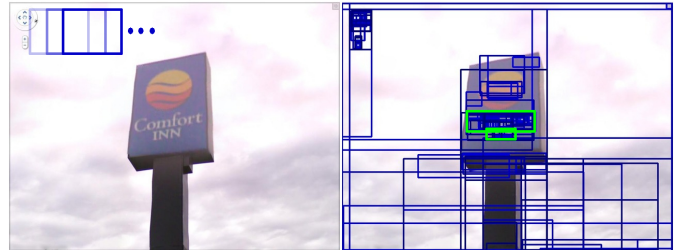
Fig. 1: Sliding a window for all possible locations, sizes, and aspect ratios represents a considerable waste of resources. The best ranked 250 proposals generated with our text specific selective search method provide 100% recall and high-quality coverage of words in this particular image.

I. INTRODUCTION

Scene Text understanding consists in determining whether a given image contains textual information and if so, localizing it and recognizing its written content. Traditionally this challenging task has been tackled with a multistage pipeline where text detection, extraction, and recognition steps have been treated separately as isolated problems. More recently, an alternative framework has been proposed motivated by the high accuracy of methods for whole word recognition and the emergent use of Object Proposal techniques. This new framework has produced the best performing state-of-the-art methods for scene text end-to-end word spotting [1], [2].

Object Proposals is a recent computer vision technique for generation of high quality object locations. The main interest of such methods is their ability to speed up recognition pipelines that make use of complex and expensive classifiers by considering only a few thousands of bounding boxes. It therefore constitutes an alternative to exhaustive search, which has many well known drawbacks, and enables the efficient use of more powerful classifiers by greatly reducing the search space as shown in Figure 1.

In the context of scene text understanding, whole-word recognition methods [3], [4] have demonstrated great success in difficult tasks like word spotting or text based retrieval, however they are usually based in expensive techniques. In this scenario the underlying process is similar to the one in multiclass object recognition. It is therefore suggestive for the use of Object Proposals techniques mimicking the state of the art object recognition pipelines.

Traditionally, high precision specialized detectors have been used for segmentation of text in natural scenes, and afterwards text recognition techniques applied to their output [5].

But it is a well known fact that the perfect text detector, able to work in any conditions, does not exist up to date. In fact, to mitigate the lack of a perfect detector Bissacco *et al.* [6] propose an end-to-end scene text recognition pipeline using a combination of several detection methods running in parallel. Demonstrating that if you have a robust recognition method at the end of the pipeline the most important thing in earlier stages is to achieve high recall while precision is not so critical.

The dilemma is thus to choose between having a small set of detections with very high precision but most likely losing some of the words in the scene, or a larger set of proposals, usually in the range of few thousands, with better coverage and then let the recognizer to make the final decision. The later seems to be a well-grounded procedure in the case of word-spotting and retrieval for various reasons. First, as said before, we have powerful whole-word recognizers but they are complex and expensive, second, the recall of current text detection methods may limit their accuracy, and third, sliding window can not be considered an efficient option mainly because words do not have a constrained aspect ratio.

In this paper we explore the applicability of Object Proposals techniques in scene text understanding, aiming to produce a set of word proposals with high recall in an efficient way. We propose a simple text specific selective search strategy, where initial regions in the image are grouped by agglomerative clustering in a hierarchy where each node defines a possible word hypothesis. Moreover, we evaluate different state of the art Object Proposals methods in their ability of detecting text words in natural scenes. We compare the proposals obtained with well known class-independent methods with our own method, demonstrating that our proposal is superior in its ability of producing good quality word proposals in an efficient way.

---

[1] http://github.com/lluisgomez/TextProposals

II. Related Work

The use of Object Proposals methods to generate candidate class-independent object locations has become a popular trend in computer vision in recent times. A comprehensive survey can be found in Hosang *et al.* [7]. In general terms, we can distinguish between two major types of Object proposals methods: the ones that make use of exhaustive search to evaluate a fast to compute objectness measure [8], [9], [10], and the ones where the search is segmentation-driven [11], [12], [13].

In the first category, Alexe *et al.* [8] propose a generic objectness measure for a given image window that combines several image cues, such as a saliency score , the color contrast to its immediate surrounding area, the edge density, and the number of straddling contours. Computation of these features is made efficient by using integral images. Cheng *et al.* [9] propose a very fast objectness score using the norm of image gradients in a sliding window, with a suitable resizing of windows into a small fixed size. A different sliding window driven approach is given by Zitnick *et al.* [10], where a box objectness score is measured as the number of edges [14] that are wholly contained in the box minus those that are members of contours that overlap the box's boundary. Using efficient data structures they manage to evaluate millions of candidate boxes in a fraction of second.

On the other hand, selective search methods make use of image's inherent structure through segmentation to guide the search. In this spirit, Gu et al. [15] make use of a hierarchical segmentation engine [16] and consider each node in the hierarchy as an object part hypothesis. Uijlings et al. [11] argue that a single segmentation and grouping strategy is not enough to generate high quality object locations in any conditions, and thus propose a selective search algorithm that uses multiple complementary strategies. In particular, they start from superpixels using different parameter settings [17] for a variety of color spaces, and then produce a set of hierarchies by merging adjacent regions using different complementary similarity measures. Another method based on superpixels merging is due to Manen *et al.*[12], using the connectivity graph induced by the segmentation [17] of an image, with edge weights representing the likelihood that two neighboring pixels belong to the same object, their Randomized Prim's algorithm generate proposals by sampling random partial spanning trees with large expected sum of weights. Finally, Krähenbühl *et al.* [13] compute an oversegmentation of the image using a fast edge detector [14] and the Geodesic K-means algorithm [18]. Then they identify a small set of seed superpixels, aiming to hit all objects in the image, and object proposals are identified as critical level sets of the Geodesic Distance Transforms (SGDT) computed for several foreground and background masks for these seeds.

The use of Object Proposals techniques in scene text understanding has been exploited very recently in two state-of-the-art word-spotting methods [1], [2] while in a distinct manner. In our previous work [1] we propose a text specific selective search method adopting a similar strategy to the selective search of Uijlings *et al.* [11] and a holistic word recognition method based on Fisher Vector representations. On the other hand, Jaderberg *et al.* [2] opt for the use of a generic Object Proposals algorithm [10] and deep convolutional neural networks for recognition.

The method proposed in this paper builds on top of our previous work [19], [20], [1], where initial regions in the image are grouped by agglomerative clustering, using complementary similarity measures, in hierarchies where each node defines a possible word hypothesis. But differs from it in two main aspects: First, we do not rely in a classifier to make strong decisions to discriminate text groups from not-text groups, second, we do not combine the different cues in any way.

III. Text Specific Selective Search

Our method is based on the fact that text, independently of the script in which it is written, emerges always as a group of similar atomic objects. We make use of the perceptual organisation framework presented in [19], where a set of complementary grouping cues are used in parallel to generate hierarchies in which each node correspond to a text-group hypotheses. Our algorithm is divided in three main steps: segmentation, creation of hypotheses through bottom-up clustering, and ranking.

In the first step we use the Maximally Stable Extremal Regions (MSER) algorithm [21] to obtain the initial segmentation of the image, as it is proven to be an efficient method for detecting text parts [22].

A. Creation of hypotheses

The grouping process starts with a set of regions $\mathcal{R}_c$ extracted with the MSER algorithm. Initially each region $r \in \mathcal{R}_c$ starts in its own cluster and then the closest pair of clusters (A, B) is merged iteratively, using the single linkage criterion (SLC) ($\min \{ d(r_a, r_b) : r_a \in A, r_b \in B \}$), until all regions are clustered together ($C \equiv \mathcal{R}_c$). Where $d(r_a, r_b)$ is a distance metric that will be explained next.

Similarly to [11] we assume that there is no single grouping strategy that is guaranteed to work well in all cases. Thus, our basic agglomerative process is extended with several diversification strategies in order to ensure the detection of the highest number of text regions in any case. First, we extract regions separately from different color channels (i.e. Red, Green, Blue, and Gray) and spatial pyramid levels. Second, on each of the obtained segmentations we apply SLC using different complementary distance metrics:

$$d^{(i)}(r_a, r_b) = (f^i(r_a) - f^i(r_b))^2 + (x_a - x_b)^2 + (y_a - y_b)^2 \quad (1)$$

where the term $\{(x_a - x_b)^2 + (y_a - y_b)^2\}$ is the squared Euclidean distance between the centers of regions r_a and r_b, and $f(r)$ is a feature aimed to measure the similarity of two regions. Our f^i features are designed to exploit the strong similarity of text regions belonging to the same word. We make use of the following simple features with low computation cost: mean gray value of the region, mean gray value in the immediate outer boundary of the region, region's major axis, mean stroke width, and mean of the gradient magnitude at the region's border.

B. Ranking

Once we have created our similarity hierarchies each one providing a set of text group hypotheses, we need an efficient way to sort them in order to provide a ranked list of proposals prioritizing the best hypotheses. In the experimental section we explore the use of the following rankings:

1) Pseudo-random ranking: We make use of the same ranking strategy proposed by Uijlings *et al.*in [11]. Particularly, each hypothesis is assigned with an increasing integer value, starting from 1 for the root node of a hierarchy and subsequently incrementing for the rest of the nodes up to the leaves of the tree. Then each of this values is multiplied with a random number between zero and one, thus providing a ranking that is randomly produced but prioritizes larger regions. As in [11] the ranking process is performed before removing duplicate hypotheses. This way if a particular grouping has been found several times within the different hierarchies, indicating a more consistent hypothesis under different similarity cues, this group is going to have more probabilities to be ranked in the top of the list.

2) Cluster meaningfulness ranking: Instead of assigning an increasing value prioritizing larger groups, we propose here to use a cluster quality measure, based on the principle of non-accidentalness, that has been proposed in [23] for hierarchical clustering validity assessment. In our case, given one of the grouping cues described in section III-A, equation 1 defines a feature space in which individual regions are projected, and the meaningfulness of a group of regions G can be calculated as the inverse of the probability of such a group being a realization of the uniform random distribution:

$$\mathcal{NFA}(G) = \mathcal{B}_G(k, n, p) = \sum_{i=k}^{n} \binom{n}{i} p^i (1-p)^{n-i} \quad (2)$$

where k is the number of regions in G, n is the total number of regions extracted from the image, and p is the ratio of the volume defined by the distribution of the feature vectors of the regions in G with respect to the total volume of the feature space. Intuitively this value is going to very small for groups comprising a set of very similar regions, that are densely concentrated in small volumes of the feature space. This measure is thus well indicated in the case of measuring text-likeliness of groups because such a strong similarity property is expected to be found in text groups. However, the ranking provided by calculating 2 in each node of our hierarchies is going to prioritize large text groups, e.g. paragraphs, rather that individual words, and thus we combine the ranking provided by equation 2 with a random number between zero and one as done before, providing a pseudo-random ranking where more meaningful hypothesis are prioritized.

3) Text classifier confidence: Finally, we propose the use of a weak classifier to generate our ranking. The basic idea here is to train a classifier to discriminate between text and non-text hypotheses and to produce a confidence value that can be used to rank group hypotheses. Since the classifier is going to be evaluated on every node of our hierarchies, we aim to use a fast classifier and features with low computational cost. We train a Real AdaBoost classifier with decision stumps using as

features the coefficients of variation of the individual region features f^i described in section III-A: $F^i(G) = \sigma^i/\mu^i$, where μ^i and σ^i are respectively the mean and standard deviation of the region features f^i in a particular group G, $\{f^i(r) : r \in G\}$. Intuitively the value of F^i is smaller for text hypotheses than for non-text groups, and thus the classifier would be able to generate a ranking prioritizing the best hypotheses. Notice that all F^i group features can be computed efficiently in an incremental way along the SLC hierarchies, and that all f^i region features have been previously computed.

IV. EXPERIMENTS AND RESULTS

In our experiments we make use of two standard scene text datasets: the ICDAR Robust Reading Competition dataset (ICDAR2013) [24] and the Street View Dataset (SVT) [25]. In both cases we provide results for their test sets, consisting in 233 and 249 images respectively, using the original word level ground-truth annotations.

The evaluation framework used is the standard for Object Proposals methods [7] and is based on the analysis of the detection recall achieved by a given method under certain conditions. Recall is calculated as the ratio of GT bounding boxes that have been predicted among the object proposals with an intersection over union (IoU) larger than a given threshold. This way, we evaluate the recall as a function of the number of proposals, and the quality of the first ranked N proposals by calculating their recall at different IoU thresholds.

A. Evaluation of diversification strategies

First, we analyse the performance of different variants of our method by evaluating the combination of diversification strategies presented in Section III. Table I shows the average number of proposals per image, recall rates, and time performance obtained with some of the possible combinations. We select two of them, that we will call "FAST" and "FULL" as they represent a trade-off between recall and time complexity, for further evaluation.

Method	# prop.	0.5 IoU	0.7 IoU	0.9 IoU	time(s)
I+D	536	0.84	0.65	0.41	0.26
I+DF	993	0.91	0.78	0.53	0.29
I+DFBGS	1323	0.95	0.86	0.60	0.45
RGB+DF	3359	0.96	0.91	0.69	0.73
RGBI+DFBGS	5659	0.98	0.94	0.75	1.72
P2+RGBI+DFBGS	8164	0.98	0.96	0.79	2.18

TABLE I: Max recall at different IoU thresholds and running time comparison of different diversification strategies in the ICDAR2013 dataset. We indicate the use of individual color channels: (R), (G), (B), and (I); spatial pyramid levels: (P2); and similarity cues: (D) Diameter, (F) Foreground intensity, (B) Background intensity, (G) Gradient, and (S) Stroke width.

B. Evaluation of proposals' rankings

Figure 2 shows the performance of our "FAST" pipeline at 0.5 IoU using the various ranking strategies discussed in Section III. The area under the curve (AUC) is 0.39 for NFA, 0.43 both for PR and PR-NFA rankings, while a slightly better 0.46 for the ranking provided by the weak classifier. Since

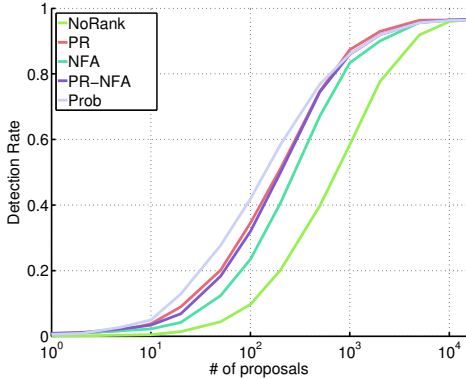the overhead of using the classifier is negligible we use this ranking strategy for the rest of the experiments.

Fig. 2: Performance of our "FAST" pipeline at 0.5 IoU using different ranking strategies: (PR) Pseudo-random ranking, (NFA) Meaningfulness ranking, (PR-NFA) Randomized NFA ranking, (Prob) the ranking provided by the weak classifier.

C. Comparison with state of the art

In the following we further evaluate the performance of our method in the ICDAR2013 and SVT datasets, and compare it with the following state of the art Object Proposals methods: BING [9], EdgeBoxes [10], Randomized Prim's [12] (RP), and Geodesic Object Proposals [13] (GOP).

In our experiments we use publicly available code of these methods with the following setup. For BING we use the default parameters: base of 2 for the window size quantization, feature window size of 8×8, and non maximal suppression (NMS) size of 2. For EdgeBoxes we also use the default parameters: step size of the sliding window of 0.65, and NMS threshold of 0.75; but we change the max number of boxes to 10^6. GOP is configured with Multi-Scale Structured Forests for the segmentation, 150 seeds heuristically placed, and 8 segmentations per seed; in this case we tried other configurations in order to increase the number and quality of the proposals without success. For RP we use the default configuration with 4 color spaces (HSV,Lab,Opponent,RG) because it provided much better results than sampling from a single graph, while being 4 times slower.

Tables II and III show the performance comparison of all the evaluated methods in ICDAR2013 and SVT datasets respectively. A more detailed comparison is provided in Figure 3. All time measurements in Tables II and III have been calculated by executing code in a single thread on the same i7 CPU for fair comparison, while most of them allow parallelization. For instance the multi-threaded version of our method is able to achieve execution times of 0.31 and 0.71 seconds respectively for the "FAST" and "FULL" variants in the ICDAR2013 dataset.

As can be seen in Table II and Figure 3 our method outperforms all the evaluated algorithms in terms of detection recall on the ICDAR2013 dataset. Moreover, it is important to notice that detection rates of all the generic Object Proposals heavily deteriorate for large IoU thresholds while our text

Method	# prop.	0.5 IoU	0.7 IoU	0.9 IoU	time(s)
BING [9]	2716	0.63	0.08	0.00	1.21
EdgeBoxes [10]	9554	0.85	0.53	0.08	2.24
RP [12]	3393	0.77	0.45	0.08	12.80
GOP [13]	855	0.45	0.18	0.08	4.76
Ours-FAST	3359	**0.96**	**0.91**	**0.69**	**0.79**
Ours-FULL	8164	**0.98**	**0.96**	**0.79**	2.25

TABLE II: Average number of proposals, recall at different IoU thresholds, and running time comparison with Object Proposals state of the art algorithms in the ICDAR2013 dataset.

specific method provides much more stable rates indicating a better coverage of text objects, see the high AUC difference in Figure 3 bottom plots.

Method	# prop.	0.5 IoU	0.7 IoU	0.9 IoU	time(s)
BING [9]	2987	0.64	0.09	0.00	0.81
EdgeBoxes [10]	15319	0.94	**0.63**	0.04	2.71
RP [12]	5620	0.02	0.00	0.00	10.51
GOP [13]	778	0.53	0.19	0.03	4.31
Ours-FAST	3791	0.90	0.46	0.03	**0.66**
Ours-FULL	10365	**0.95**	0.61	**0.06**	2.22

TABLE III: Average number of proposals, recall at different IoU thresholds, and running time comparison with Object Proposals state of the art algorithms in the SVT dataset.

The results on the SVT dataset in Table III and Figure 3 exhibit a radically distinct scenario. While our "FULL" pipeline is slightly better than EdgeBoxes at 0.5 IoU, the later is able to outperform both of our pipelines at 0.7 and our "FAST" variant at 0.5. Moreover, in this dataset our method does not provide the same stability properties shown before. This can be explained because both datasets are very different in nature, SVT contains more challenging text, with lower quality and many times under bad illumination conditions, while in ICDAR2013 text is mostly well focussed and flatly illuminated. Still, the AUC in most of the plots in Figure 3 show a fairly competitive performance for our method.

V. CONCLUSIONS

In this paper we have evaluated the performance of generic Object Proposals in the task of detecting text words in natural scenes. We have presented a text specific method that is able to outperform generic methods in many cases, or to show competitive numbers in others. Moreover, the proposed algorithm is parameter free and fits well the multi-script and arbitrary oriented text scenario.

An interesting observation of our experiments is that while in class-independent object detection generic methods suffice with near a thousand proposals to achieve high recalls, in the case of text we still need around 10000 in order achieve similar rates, indicating there is a large room for improvement in specific text Object Proposals methods.
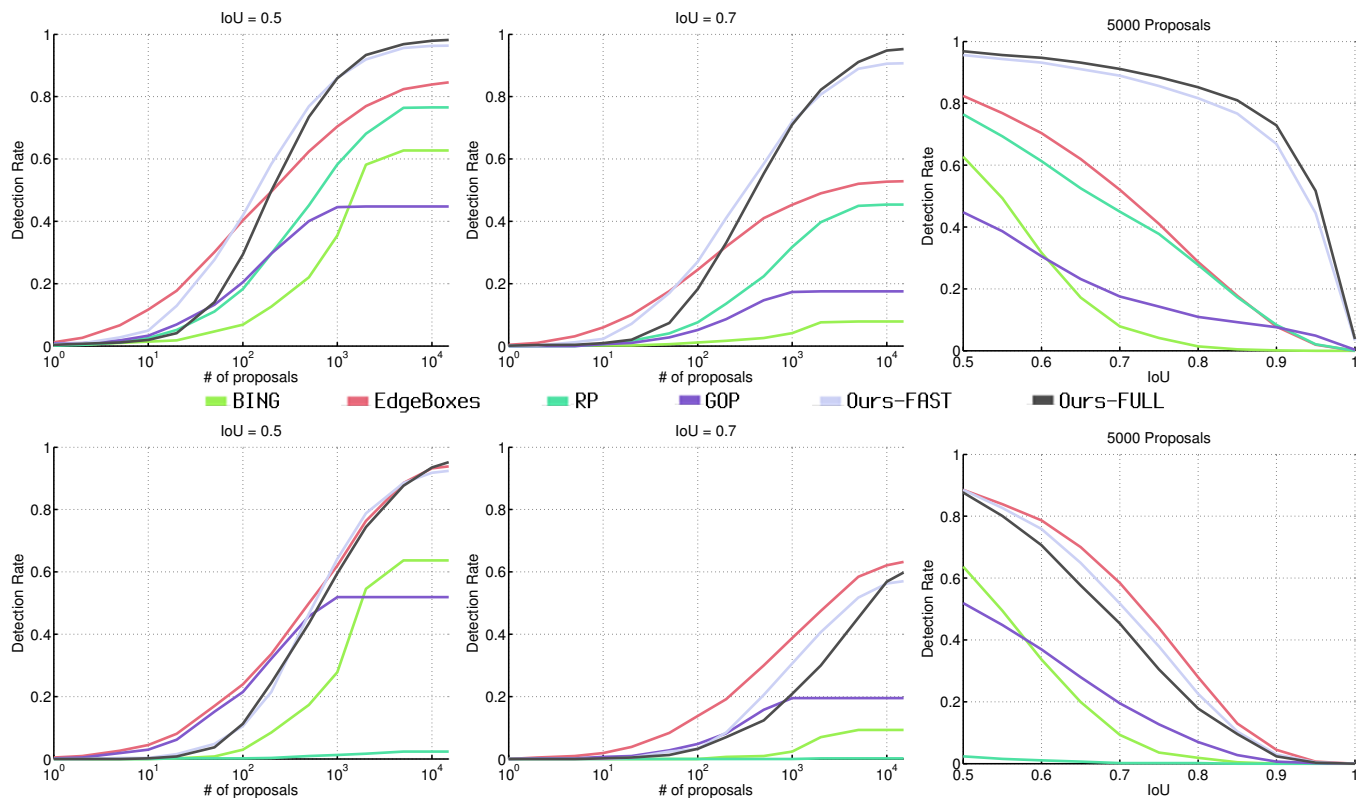
VI. ACKNOWLEDGEMENTS

Fig. 3: A comparison of various state-of-the-art object proposals methods in the ICDAR2013 (top) and SVT (bottom) datasets. (left and center) Detection rate versus number of proposals for various intersection over union thresholds. (right) Detection rate versus intersection over union threshold for various fixed numbers of proposals.

REFERENCES

[1] J. Alamazan, S. Ghosh, L. Gomez, E. Valveny, and D. Karatzas, "A selective search framework for efficient end-to-end scene text recognition and retrieval," *Conference paper under review*, 2015. 1, 2

[2] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Reading text in the wild with convolutional neural networks," *arXiv preprint arXiv:1412.1842*, 2014. 1, 2

[3] V. Goel, A. Mishra, K. Alahari, and C. Jawahar, "Whole is greater than sum of parts: Recognizing scene text words," in *ICDAR*, 2013. 1

[4] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," in *TPAMI*, 2014. 1

[5] L. Gomez and D. Karatzas, "Scene text recognition: No country for old men?" in *IWRR - ACCVW*, 2014. 1

[6] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "Photoocr: Reading text in uncontrolled conditions," in *ICCV*, 2013. 1

[7] J. Hosang, R. Benenson, and B. Schiele, "How good are detection proposals, really?" in *BMVC*, 2014. 2, 3

[8] B. Alexe, T. Deselaers, and V. Ferrari, "Measuring the objectness of image windows," *TPAMI*, 2012. 2

[9] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *CVPR*, 2014. 2, 4

[10] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *ECCV*, 2014. 2, 4

[11] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *IJCV*, 2013. 2, 3

[12] S. Manen, M. Guillaumin, and L. V. Gool, "Prime object proposals with randomized prim's algorithm," in *ICCV*, 2013. 2, 4

[13] P. Krähenbühl and V. Koltun, "Geodesic object proposals," in *ECCV*, 2014. 2, 4

[14] P. Dollár and C. L. Zitnick, "Structured forests for fast edge detection," in *ICCV*, 2013. 2

[15] C. Gu, J. J. Lim, P. Arbeláez, and J. Malik, "Recognition using regions," in *CVPR*, 2009. 2

[16] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *TPAMI*, 2011. 2

[17] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *IJCV*, 2004. 2

[18] F. Perazzi, P. Krähenbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *CVPR*, 2012. 2

[19] L. Gomez and D. Karatzas, "Multi-script text extraction from natural scenes," in *ICDAR*, 2013. 2

[20] ——, "A fast hierarchical method for multi-script and arbitrary oriented scene text extraction," *arXiv preprint arXiv:1407.7504*, 2014. 2

[21] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and Vision Computing*, 2004. 2

[22] L. Neumann and J. Matas, "Text localization in real-world images using efficiently pruned exhaustive search," in *Proc. ICDAR*, 2011. 2

[23] F. Cao, J. Delon, A. Desolneux, P. Musé, and F. Sur, "An a contrario approach to hierarchical clustering validity assessment," INRIA, Tech. Rep., 2004. 3

[24] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez, S. Robles, J. Mas, D. Fernandez, J. Almazan, and L. P. de las Heras, "Icdar 2013 robust reading competition," in *ICDAR*, 2013. 3

[25] K. Wang and S. Belongie, "Word spotting in the wild," in *ECCV*, 2010. 3