Exploiting Colour Information for Better Scene Text Recognition

Muhammad Fraz[1]
M.Fraz@lboro.ac.uk
M. Saquib Sarfraz[2]
Muhammad.Sarfraz@kit.edu
Eran A. Edirisinghe[1]
E.A.Edirisinghe@lboro.ac.uk

[1] Department of Computer Science
Loughborough University
Loughborough, UK

[2] Computer Vision for Human Computer Interaction Lab
Karlsruhe Institute of Technology
Karlsruhe, Germany

Abstract

This paper presents an approach to text recognition in natural scene images. The main contribution of this paper is the efficient exploitation of colour information for the identification of text regions in the presence of surrounding noise. We propose a pipeline of image processing operations involving the bilateral regression for the identification of characters in the images. A pre-processing step has been proposed to increase the performance of bilateral regression based character identification. The proposed method segments the characters so well that a combination of an off the shelf feature representation and classification technique achieves state-of-the-art character recognition performance. The capability of the framework is further extended for word recognition where a basic string matching criterion is applied to match recognized strings against a lexicon to eliminate errors that occur due to inaccuracies in character classification. We performed extensive experiments to evaluate our method on challenging datasets (Chars74K, ICDAR03, ICDAR11 and SVT) and results show that the proposed method superseded the existing state-of-the-art techniques.

1 Introduction

The problem of scene text recognition has gained significant importance because of its numerous applications. A variety of methods has been recently proposed that explore various theoretical and practical aspects to solve this problem. In this work, we focus towards a framework to recognize the text present in outdoor scene images. The text information carries one important property, that is, its colour in comparison to its background. Text information is always placed in such a way that it stands out from its background. In the same way, most of the time the characters in a word possess similar colour that helps us to recognize the letters of a particular word. We exploit this characteristic of text regions to solve the problem of character recognition. The character recognition pipeline is further extended in to a word recognition framework where the estimated word combinations are matched against a lexicon.

The existing approaches for scene text recognition can be roughly divided in to two broad categories: Region grouping based methods and object recognition based methods. The methods in first category usually relies on binarization strategy and use an off the shelve Optical Character Recognition (OCR) engine to recognize the segmented text. The binarization technique works great in document text but is vulnerable to noise and low resolution that makes it less suitable for scene text recognition task. An advantage of such techniques is that they are computationally inexpensive due to which they possess high feasibility for practical systems where real time performance is crucial. On the other hand, the object recognition based methods assume character recognition a similar task as object recognition with high intra-class variation. These methods rely on discriminative shape representation features and classifiers to recognize the characters. These approaches usually contain a second stage where they perform word recognition. The word recognition stage makes use of complex post processing procedures to finalize the word recognition from a large number of candidate detections. These techniques are less prone to factors like noise and low resolution but their disadvantage is that they involve complex and computationally expensive procedures which make these techniques less suitable for practical systems.

In this work, we have combined region grouping method with object recognition based strategy to achieve the advantages of both techniques. First, we binarize the image using colour information and perform foreground segmentation to separate characters from background. Next, we extract shape representation features on binary images and perform character classification using a pre-trained classifier. The recognized characters form words that are fed in to a string similarity matching stage where lexicon based search is performed to find the closest matching word.

The proposed characters recognition pipeline outperforms the current state-of-the-art by a significant margin of 8% on Chars74k [1] dataset. In the same way, the proposed word recognition pipeline outperforms the state-of-the-art on challenging ICDAR03-Word [17], ICDAR11-Word [13] benchmark datasets.

The rest of the paper is organized as follows: In section 2, we briefly overview the recent work for character and word recognition in the wild. The proposed framework is presented in detail in section 3. The results and discussion about experimental evaluation is presented in section 4 followed by the conclusion in section 5.

2 Related Work

A significant volume of literature exists that deals with the problem of character and word recognition in natural scene images. A number of specialized feature representations, binarization techniques, segmentation methods and word models have been proposed to-date, yet the problem of text recognition is open because of diversified nature of text and the presence of high inter-class similarity along with high intra-class variation. In this section, we briefly cover the most recent work in character recognition and word understanding.

In character recognition, Campos et al. [1] introduced Chars74k dataset of characters collected from natural scene images. They showed that commercial OCR engines do not achieve good accuracy scores for natural scene character images and therefore they proposed a Multiple Kernel Learning based method. Wang et al. [19] proposed the use of Histogram of Oriented Gradient (HOG) [2] features together with Nearest Neighbour classifier and showed the improved performance. They further enhanced their work in [20] using a Bayesian inference based method and showed significant improvement on ICDAR-CH dataset. Sheshdari

et al. [14] used HOG features with exemplar Support Vector Machine (SVM) and affine warping to raise the performance bar by a considerable value. Yi et al. [23] compared local and global HOG features for scene character recognition. A few word recognition and end-to-end scene text recognition methods [15], [5] have also reported character recognition scores separately. The most recent work in character recognition has been proposed by Lee et al. [5], their proposed approach use discriminative region based feature pooling to automatically learn the most informative sub-regions of each scene character within a multi-class classification framework achieving the state-of-the-art performance.

In terms of word recognition, a number of approaches have been recently emerged that focus on specialized module for word recognition. Smith et al. [16] proposed a similarity expert algorithm to remove the logical inconsistencies in an equivalence graph and perform search for the maximum likelihood interpretation of a sign as an integer programming. The work in [15],[9],[7] built Conditional Random Field (CRF) models on the potential character locations in a sliding window based search and add the linguistic knowledge and spatial constraints to compute pairwise costs for word recognition. The work in [19],[20] used pictorial structures to detect words in the image. The pictorial structures find an optimal configuration of a particular word using the scores and locations of detected characters. Weinmann et al. [21] formulated Markovian model score on the segmented candidates on the basis of appearance, geometric and linguistic characteristics. Neuman et al. [11] proposed a real time text recognition system where they extract candidate character regions using a set of Extremal Regions (ERs) and then perform exhaustive search with feedback loops to group ERs into words and recognize them in an OCR stage that is trained by using synthetic fonts. Recently, Yao et al. [22] proposed a new feature representation technique named as Strokelets, that captures the essential substructures of characters at different granularities.

Apart from word recognition module, various approaches have been recently proposed for character identification in images such as Connected Components (CCs) [12], Binarization, [21],[6], ERs [11], Graph-cuts [8], Sliding Windows [20] and k-Mean clustering [18]. Recently, Field et. al. [3] proposed Bilateral Regression for text segmentation, it uses colour clustering as a starting point to fit a regression model for each image and separate foreground pixels from background ones using an error threshold. The method reports a superior performance in comparison to existing segmentation techniques.

3 Proposed Framework

Our framework comprises of three main stages: Character Identification, Character Recognition and Word Recognition. Each stage is separately explained in this section.

3.1 Character Identification

The key requirement for any character identification technique is not only to segment the characters from background but to segment them in such a fine way that they remain separated from each other. The conventional CC based methods do not perform well. Most of the recently proposed techniques use sliding window based approach for this purpose. This however generates too many candidates and requires a large number of evaluations. Following [3], we use the bilateral regression to segment characters. However, our approach is different than the original method in that we only use it to estimate the horizontal location

Figure 1: Improved character identification. Row 1 shows the original images. Row 2 shows the results of character segmentation using Bilateral Regression. Row 3 shows the results of character segmentation using the combination of proposed pre-processing and Bilateral Regression.

of each character in word image. Our objective is the estimation of the starting column and width of each character in the word image.

The bilateral regression models the foreground pixels by using a weighted regression that assigns weight to each pixel according to its location with respect to foreground in feature space. The pixels that belong to the foreground get high weights in comparison to the pixels belonging to background. In this case, the regression model in equation 1 represents the quadratic surface that best models the image as a function of pixel locations.

$$z = ax^2 + by^2 + cxy + dx + ey + f \tag{1}$$

The error between each pixel in the image and the model is computed. The pixels with error value higher than a specific threshold are excluded as background while the remaining pixels are finalized as foreground pixels. The bilateral regression requires modelling the top 'n' colours in each image separately. In a post processing procedure it then chooses the segmentation that is most likely to contain the foreground text by comparing the shape descriptors of foreground regions with a training set. This makes the overall process complex and leads to various false foreground segmentation results.

We enhance the operation of bilateral regression by a pre-processing step where the foreground colour is estimated a priori. We apply n-level colour quantization to achieve binary image for each quantization level. We use Minimum Variance Quantization (MVQ) originally proposed by Heckbert [4].The quantization process merges all the noisy areas together at different quantization levels while separately clusters the pixels of text regions from their surrounding pixels. The process is extremely fast and accumulates similar coloured regions very well even in the presence of noise. Here, keeping in view the relatively large variance of colours in the whole word as opposed to the individual character, we quantize each word image in to three colours and analyse the respective binary maps for three quantization levels to estimate the foreground. The main motivation is the observation that in the detected word, background (having the uniform colour) is captured in one large colour cluster. While most of the foreground character pixels gather in another cluster. The small variations typically present along the edges of the characters due to noise and illuminations are captured in another small cluster. The pixels in each colour cluster are assigned a value of 1 (white) to generate the respective binary map. The binary map that contains the highest number of white pixels along the border is declared as the background information and is dropped straight away. From the remaining two binary maps, we check the number of white pixels along the borders as well as the total number of white pixels within that binary map. The binary map that contains less white pixels is dropped and the remaining binary map is finalized as the foreground map. The cluster centre of the quantized colour of the pixels against

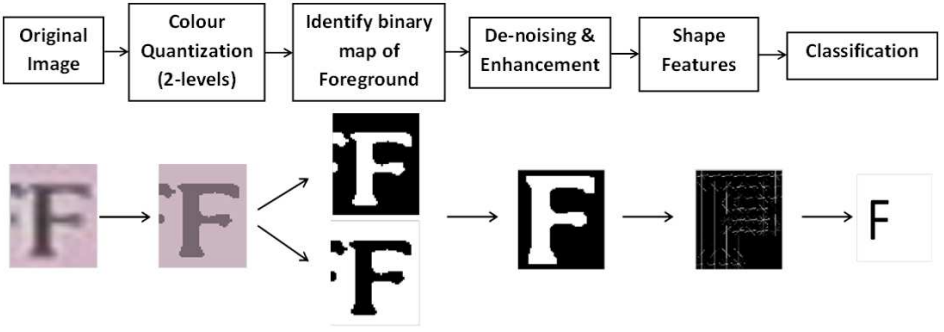


Figure 2: The block diagram of the character recognition framework with the visualization of output at each stage.

the foreground binary map is then used to perform bilateral regression.

The characters are cropped from the actual word images using the estimated horizontal location and width from bilateral regression while the height is kept same as the height of the actual word image. In this way, we might get some background information but the chances of segmenting only a part of character are reduced. Figure 1 depicts the improvement in segmentation we get using the proposed pre-processing procedure in bilateral regression. Finally, the segmented masks are used to crop the characters from original (coloured) image and fed into the character recognition pipeline explained next.

3.2 Character Recognition

Accurate character recognition has a significant importance to achieve correct word recognition. After a character is properly segmented the recognition requires being robust with regards to the noise, illumination and perspective distortions. Figure 2 shows all the important steps for character recognition. This section describes each step in detail.

Foreground Segmentation: Consider an image containing a character along with background noise. Similar to the character identification stage, we use colour quantization to enhance the character. Note that the foreground binary mask obtained for each character in the previous stage cannot be directly used. Although it gives a good separation in terms of segmenting the character from the detected text the character itself may not be well represented because of the missing edge pixels. We found on the basis of extensive experimentation that for a character image 2-level quantization is good enough to recover the full character pixels from background. We therefore generate two binary images corresponding to the two colour levels by assigning the pixels for each colour cluster a value '1' (white similar to the previous stage), we categorize the two binary images as foreground character map by simply analysing the white pixels density along the borders of each binary map. The binary map for the background tends to have more white pixels near the borders of the image as compared to the binary map of foreground character that is less likely to have high white pixel density around the borders especially at the corners. We use this property to classify the binary maps as foreground and background. A 5-by-5 pixels window from each of the four corners of the binary map is selected and the total number of corner white pixels in each binary map are counted. The binary map that possesses the higher total number of corner white pixels is

Figure 3: A few challenging examples of character images from ICDAR03-CH and Chars74k datasets where proposed method extracts and enhances the binary map of character region. (a) Input Images. (b) Quantized Images with 2-Colors MVQ. (c) Binary map of characters after noise reduction and enhancement.

considered as background and the other binary map is classified as the character map.

Noise Reduction and Enhancement: The foreground binary map is fed in to a noise reduction and enhancement stage where most of the unwanted pixels are removed. Morphological closing, spur removal and dilation operations are applied to remove noisy pixels and enhancing the pixels belonging to the character region. Afterwards, we find the bounding boxes around the CCs and the biggest CC in the binary map is cropped as character map. The binary map of the character is resized to p-by-q pixels and padded with an array of five black pixels at all sides resulting in a binary image of size (p+10)-by-(q+10) pixels with character map centred in it. The value of 'p' and 'q' is empirically selected as '64' and '48' respectively. It is observed that the characters are slightly taller than their width, therefore, the selected size maintains the aspect ratio of characters. Figure 3 shows a few examples where the proposed colour based binarization scheme segments and enhances the character binary map.

Feature Extraction and Classification: The binary map acquired in the previous step is directly used for feature extraction. We compute HOG features on the binary map of the character and use this representation to classify the character. For classification, we use a multi-class SVM. In this paper, we consider Digits (10 classes) and English letters (52 Classes) i.e. the alphabets ζ = [0,...,9;A,...,Z;a,...,z] and $|\zeta|$=62. Hence, a 62-class non-linear SVM is trained in a one-vs-all manner. The best parameters for training the SVM model have been learned using Radial Basis Function (RBF) kernel with 5-fold cross validation.

3.3 Word Recognition:

The word recognition stage requires the segmentation of characters in a cropped word image and then recognition of each character with a reasonable accuracy to form the correct word. The inaccuracies in character segmentation and recognition primarily lead to false alarms. Since, we have proposed improvements in both prior stages, therefore, here we are relying on a simple lexicon based alignment procedure to remove errors that occur in character recognition. The errors in character recognition are inevitable because of high interclass similarity between various characters i.e. 'I' and 'l', '0' and 'O' etc. The alignment is performed using Lavenshtien distance.

Alignment with Lexicon: The character recognition pipeline predicts the character label 'l' on the basis of highest probability estimate but in order to deal with wrong recognitions

we cannot rely on only the first predicted label especially if the probability estimate for that recognition is too low. To deal with this, we select top η predicted labels based on a confidence score S_c. The confidence score is the sum of probability of top η predicted labels. The value of η varies in every test case. When S_c reaches to a threshold 'τ' the character labels corresponding to those probability estimates are included in the predicted word combination. We experimentally found the value τ=0.35 for our evaluation process. One potential problem that occurs in this set up is the formation of an enormous number of character combinations for certain images. In order to avoid that, we limit the total number of words by selecting only 30 words with highest sum of probability estimates of characters.

for *i= 1:len(word)* **do**
  η=1;
  Assign S_c = P(η) ; // P(η) is the Probability of the ηth predicted label.
  while *($S_c < \tau$ and $\eta \leq 30$)* **do**
    EstimatedWord(η,i) = l(η) ; // Update word combinations by including ηth label l(η).
    S_c = S_c+P(η) ; // update confidence score by adding next highest probability.
    η=η+1;
  end
end

Algorithm 1: Computing words using the combination of characters with high recognition probabilities.

Next, we find the correct word from predicted character combinations. The predicted words are aligned with the words available in the lexicon using a string similarity measure. The closest matched word in the lexicon is declared as the word in the image. A number of word models have been recently proposed to achieve better word recognition accuracy. We however, adopt a simple strategy where the Lavenshtein distance [10] between two strings is used to compare the predicted word string with the words in the lexicon. It basically computes the total number of operations (insertion, deletion, and replacement) required to align one string with the other. The word recognition pipeline is simple but fully capable of performing well provided the characters are recognized with a reasonably good accuracy.

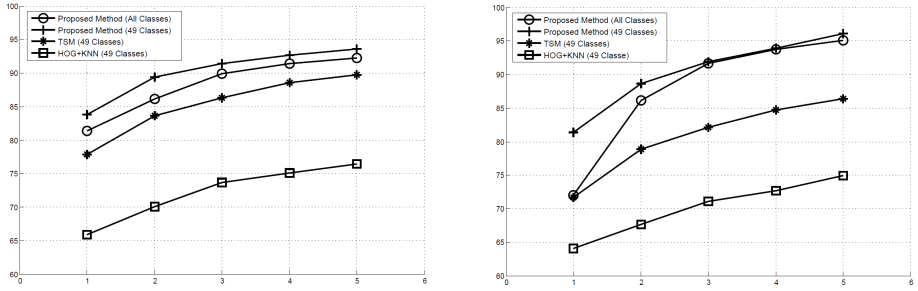
4 Experiments and Results

4.1 Character Recognition

To evaluate the performance of proposed character recognition method, we use two benchmark datasets for scene character classification task: Chars74K-15 and ICDAR03-CH dataset.

Chars74K dataset contains 12505 images of characters extracted from scene images. The images are divided into 62 classes (52 alphabets and 10 number digits). The authors have provided various training and test splits of dataset for the research community to perform fair comparison of results. Chars74K-15 is one subset that contains 930 (15 per class) training images and another 930 (15 per class) test images from Chars74K dataset.

ICDAR03-CH dataset contains 11482 character images, this does not include non al-



ICDAR03-CH Dataset Chars74k Dataset

Figure 4: Rank-Wise character recognition scores on two datasets for all (62) classes and for 49 classes. The results are compared with [15] and [5] where they reported the efficiency of their methods on 49 classes.

phanumeric characters. The dataset comprises of 6113 training and 5369 test images.

Table 1 lists the character classification results. The proposed framework outperforms existing techniques by achieving the classification accuracy of 81% on ICDAR03-CH dataset and 72% on Chars74K-15 dataset. It is evident from results that the non-informative regions around the characters have significant degradation impact on recognition accuracy. Also, the intensity variation within the strokes of characters causes a negative impact on the performance. The proposed framework not only removes the noisy background regions to achieve a nicely centred character binary map but also attenuate the effect of intensity variation within the strokes of characters. This results in a strongly discriminative shape feature extraction along the contour of characters.

The proposed methods in [15] and [5] merge similar character classes to reduce the total the number from 62 to 49. This reduces the confusion among similar character classes. We also evaluate our approach in 49-classes setup. Table 2 presents the comparison where the proposed methodology superseded the other techniques with a substantial margin especially on Chars74k-15 dataset. Figure 4 shows the rank-wise scores for top 5 probabilities of character recognition. The recognition rates for the top five candidates go up to 94% for 49 classes and 92.5% for all (62) classes in ICDAR03-CH dataset which indicates that the use of top 5 character classifications shall potentially the enhance word recognition accuracy.

4.2 Word Recognition

To evaluate the performance of word recognition frame work we used ICDAR03-WD, ICDAR11-WD and SVT-WD benchmark datasets. For a fair comparison we use the same training and testing splits as provided in [20] as well as the lexicons. A number of different sized lexicons have been provided in the work of [20]. We used 'FULL' and '50' lexicons for evaluation and comparison. 'FULL' lexicon contains all the words from test set of ICDAR03-WD dataset whereas in '50', there are 50 distractor words. Note that we use ICDAR03-CH data for training. Following [5],[9],[12],[15],[22], we skipped the words with 2 or fewer characters as well as those with non-alphanumeric characters.

The proposed pipeline achieves 89.37% and 88.94% recognition accuracy on ICDAR03 and ICDAR11 datasets respectively in small (50 words) lexicon setup and out-performs the state-of-the-art. The performance degradation in case of large lexicon is due to the increased number of distractor words. The recognition accuracy on SVT dataset is 78.66% which is

Method	ICDAR03-CH	Char74K-15
Proposed Method	**81**	**72**
MLFP [5]	79	64
GHOG+SVM [23]	76	62
LHOG+SVM [23]	75	58
HOG+NN [20]	52	58
MKL [1]	-	55
NATIVE+FERNS [20]	64	54
GB+SVM [1]	-	53
GB+NN [1]	-	47
SYNTH+FERNS [20]	52	47
SC+SVM [1]	-	35
SC+NN [1]	-	34
ABBYY [1]	21	31

Table 1: Scene character classification accuracy (%) comparison with other methods.

Method	ICDAR03-CH	Char74K-15
Proposed Method	**83**	**81**
MLFP [5]	81	74
TSM [15]	77.86	71.67
HOG+NN [15]	65.92	64.02

Table 2: Comparison of scene character classification accuracy (%) on ICDAR03 and Chars74k benchmark datasets for 49 character classes with the work in [5] and [15].

Method	ICDAR03-Full	ICDAR03-50	ICDAR11-Full	ICDAR11-50	SVT
Proposed Method	77.47	**89.37**	78.58	**88.94**	78.66
Strokelets [22]	**80.33**	88.48	-	-	75.89
MLFP+PLEX [5]	76	88	77	88	**80**
MRF [21]	-	-	-	-	78.05
TSM+CRF [15] (49*classes*)	79.30	87.44	**82.87**	87.04	73.51
Mishra et. al. [9]	67.79	81.78	-	-	73.26
Mishra et. al. [7]	-	81.78	-	-	73.26
Novikova et. al [12]	-	83	-	-	73
TSM+PLEX [15] (49*classes*)	70.47	80.70	74.23	80.25	69.51
SYNTH+PLEX [20]	62	76	-	-	57
ABBY [20]	55	56	-	-	35

Table 3: Comparison of cropped word recognition performance using recognition accuracy (%).

Figure 5: Successful word recognition results on SVT and ICDAR images.

Figure 6: Some examples where proposed technique fails.

second to the highest recognition accuracy reported to date. Considering the simplicity of word recognition pipeline, the achieved recognition accuracy is promising. Table 3 compares the cropped word recognition performance of proposed method with other techniques that used the similar experimental setup. Results clearly depict that Lavenshtein distance based alignment along with the proposed character recognition method achieves exceptional performance on standard datasets. Figure 5 shows some examples of successful word recognition.

One drawback of the proposed character identification module is that it fails in scenarios where the characters are joined either because of font or because of lighting and viewing angle. Figure 6 shows a few examples where proposed character identification technique failed that resulted in wrong word recognition output.

4.3 Computational Performance

The proposed framework is implemented in MATLAB. The average execution time for the proposed word recognition pipeline on a standard PC is 1.7 seconds. The separate average execution time for three stages: Character Identification, Character Recognition and Word Recognition is 1.2 sec., 0.4 sec. and 0.1 sec. respectively. Note that the code is unoptimized. The execution time can be further reduced near real-time with the inclusion of code optimization and parallel processing techniques.

5 Conclusion

We proposed an end-to-end scene text recognition framework. The recognition pipeline combined region grouping method with object recognition based strategy to achieve state-of-the-art performance on benchmark datasets. The proposed modification for bilateral regression based segmentation drastically improved character identification performance. The binary maps of the segmented characters have been directly used to extract shape features and fed in to the trained SVM classifier. Finally, a basic string similarity measure has been used to align the estimated words with the lexicon to remove inaccuracies. The experimental results show that proposed framework is accurate, fast, simple and exploitable for practical applications.

References

[1] T. de. Campos, B. Babu, and M. Verma. Character recognition in natural images. In *VISAPP*, 2009.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, June 2005. doi: 10.1109/CVPR.2005.177.

[3] Jacqueline L. Feild and Erik G. Learned-Miller. Improving open-vocabulary scene text recognition. In *Proceedings of the 2013 12th International Conference on Document Analysis and Recognition*, ICDAR '13, pages 604–608, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-4999-6. doi: 10.1109/ICDAR.2013.125.

[4] Paul Heckbert. Color image quantization for frame buffer display. *SIGGRAPH Comput. Graph.*, 16(3):297–307, July 1982. ISSN 0097-8930. doi: 10.1145/965145.801294.

[5] C Lee, A Bharadwaj, W Di, V Jagadeesh, and R Piramuthu. Region based discriminative pooling for scene text recognition. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, 2014.

[6] Sergey Milyaev, Olga Barinova, Tatiana Novikova, Victor Lempitsky, and Pushmeet Kohli. Image binarization for end-to-end text understanding in natural images. In *ICDAR*, pages 128 – 132, 2013. URL https://dl.dropboxusercontent.com/u/34762938/SceneTextBinarization.pdf.

[7] 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*, pages 2687–2694, June 2012. doi: 10.1109/CVPR.2012.6247990.

[8] Anand Mishra, Karteek Alahari, and C. V. Jawahar. An mrf model for binarization of natural scene text. In *ICDAR*, pages 11–16, 2011.

[9] Anand Mishra, Karteek Alahari, and Cv Jawahar. Scene text recognition using higher order language priors. In *Proceedings of the British Machine Vision Conference*, pages 127.1–127.11. BMVA Press, 2012. ISBN 1-901725-46-4. doi: http://dx.doi.org/10.5244/C.26.127.

[10] Gonzalo Navarro. A guided tour to approximate string matching. *ACM Comput. Surv.*, 33(1):31–88, March 2001. ISSN 0360-0300. doi: 10.1145/375360.375365. URL http://doi.acm.org/10.1145/375360.375365.

[11] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3538–3545, June 2012. doi: 10.1109/CVPR.2012.6248097.

[12] Tatiana Novikova, Olga Barinova, Pushmeet Kohli, and Victor Lempitsky. Large-lexicon attribute-consistent text recognition in natural images. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, ECCV'12, pages 752–765, Berlin, Heidelberg, 2012. Springer-Verlag. ISBN 978-3-642-33782-6. doi: 10.1007/978-3-642-33783-3_54. URL http://dx.doi.org/10.1007/978-3-642-33783-3_54.

[13] A Shahab, F. Shafait, and A Dengel. Icdar 2011 robust reading competition challenge 2: Reading text in scene images. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, pages 1491–1496, Sept 2011. doi: 10.1109/ICDAR.2011.296.

[14] Karthik Sheshadri and Santosh Kumar Divvala. Exemplar driven character recognition in the wild. In *British Machine Vision Conference (BMVC) 2012*, September 2012.

[15] Cunzhao Shi, Chunheng Wang, Baihua Xiao, Yang Zhang, Song Gao, and Zhong Zhang. Scene text recognition using part-based tree-structured character detection. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2961–2968, June 2013. doi: 10.1109/CVPR.2013.381.

[16] D. L. Smith, J. Field, and E. Learned-Miller. Enforcing similarity constraints with integer programming for better scene text recognition. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 0:73–80, 2011. doi: http://doi.ieeecomputersociety.org/10.1109/CVPR.2011.5995700.

[17] L. P. Sosa, S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young. Icdar 2003 robust reading competitions. In *In Proceedings of the Seventh International Conference on Document Analysis and Recognition*, pages 682–687. IEEE Press, 2003.

[18] Toru Wakahara and Kohei Kita. Binarization of color character strings in scene images using k-means clustering and support vector machines. In *ICDAR*, pages 274–278. IEEE, 2011. ISBN 978-1-4577-1350-7. URL http://dblp.uni-trier.de/db/conf/icdar/icdar2011.html#WakaharaK11.

[19] Kai Wang and Serge Belongie. Word spotting in the wild. In *European Conference on Computer Vision (ECCV)*, Heraklion, Crete, Sept. 2010. URL http://vision.ucsd.edu/~kai/grocr/.

[20] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *IEEE International Conference on Computer Vision (ICCV)*, Barcelona, Spain, 2011.

[21] Jerod J. Weinman, Zachary Butler, Dugan Knoll, and Jacqueline Feild. Toward integrated scene text reading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2):375–387, 2014. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2013.126.

[22] Cong Yao, Xiang Bai, Baoguang Shi, and Wenyu Liu. Strokelets: A learned multi-scale representation for scene text recognition. June 2014.

[23] Chucai Yi, Xiaodong Yang, and Yingli Tian. Feature representations for scene text character recognition: A comparative study. In *Proceedings of the 2013 12th International Conference on Document Analysis and Recognition*, ICDAR '13, pages 907–911, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-4999-6. doi: 10.1109/ICDAR.2013.185. URL http://dx.doi.org/10.1109/ICDAR.2013.185.