Improving Open-Vocabulary Scene Text Recognition

Jacqueline L. Feild
Department of Computer Science
University of Massachusetts Amherst
jfeild@cs.umass.edu

Erik G. Learned-Miller
Department of Computer Science
University of Massachusetts Amherst
elm@cs.umass.edu

***Abstract*—This paper presents a system for open-vocabulary text recognition in images of natural scenes. First, we describe a novel technique for text segmentation that models smooth color changes across images. We combine this with a recognition component based on a conditional random field with histogram of oriented gradients descriptors and incorporate language information from a lexicon to improve recognition performance. Many existing techniques for this problem use language information from a standard lexicon, but these may not include many of the words found in images of the environment, such as storefront signs and street signs. We avoid this limitation by incorporating language information from a large web-based lexicon of around 13.5 million words. This lexicon contains words encountered during a crawl of the web, so it is likely to contain proper nouns, like business names and street names. We show that our text segmentation method allows for better recognition performance than the current state-of-the-art text segmentation method. We also evaluate this full system on two standard data sets, ICDAR 2003 and ICDAR 2011, and show an increase in word recognition performance compared to the current state-of-the-art methods.**

I. Introduction

The area of scene text recognition focuses on the recognition of text, like street and storefront signs, in images of natural scenes. Unlike images of documents containing text, natural scene images may include unusual and artistic fonts, may vary widely in color or texture and may be captured under a variety of viewing angles and lighting conditions. These characteristics make this problem challenging, and make it difficult to directly apply existing solutions for recognizing text in documents. Despite these challenges, improving scene text recognition is important, since potential benefits include the ability to translate text in the environment into other languages and improving navigation for people with low vision.

Because of the challenging characteristics of scene text, many recent approaches for scene text recognition solve a simpler version of the problem called word spotting [1], [2], [3], [4]. This version assumes that recognized words come from a small, specialized lexicon. Other methods use a larger lexicon, but still assume that correct word labels exist in that lexicon [5]. However, the assumption that text labels must be drawn from a lexicon constrains the space of possible recognized words. This limits the utility of such approaches since text in the environment is likely to contain proper nouns and other words that will not appear in a general lexicon.

In contrast, we present an open-vocabulary word recognition system for natural images that does not require recognized words to come from a given lexicon. First, we introduce a novel regression based text segmentation technique that models smooth color changes across images and can segment text that varies in color from one part of the image to another. We combine this with a recognition component that uses a conditional random field (CRF) model with histogram of oriented gradients (HOG) descriptors. Finally, we describe an error correction step that incorporates language information from a web-based lexicon of 13.5 million words. These words are all found on web pages, so they include words not traditionally found in a lexicon, like business names and street names.

We evaluate this system on the problem of open-vocabulary word recognition using two standard data sets from recent ICDAR competitions, ICDAR 2003 [6] and ICDAR 2011 [7]. We show an increase in word recognition accuracy over the current state-of-the-art on both data sets.

To summarize, in this paper we make the following contributions:

1) We introduce a novel method for text segmentation in scene text images

2) We demonstrate a new approach to incorporating web-based language information.

3) We present an efficient system for open vocabulary word recognition using a large lexicon ($\sim$13.5 million words).

4) We show state of the art experimental results for open vocabulary word recognition on standard data sets [6], [7].

II. Related Work

There has been a lot of work in the area of scene text recognition in the past few years. An exhaustive review is beyond the scope of this paper, so we will describe the methods most closely related to our work.

Wang et al. introduced the problem of word spotting using a small fixed lexicon for scene text images [1], [2]. Since its introduction, others have also approached this problem by combining bottom-up and top down cues [3], and by using a specialized text segmentation technique to simplify recognition [8]. In addition, Wang et al. [4] used unsupervised feature learning combined with a convolutional neural network in an end-to-end system.

More recently, a Robust Reading competition was held at ICDAR 2011, highlighting several new solutions to the word recognition problem [7]. Since the distribution of a new data set for the competition, others have published methods for this

problem. Novikova et al. present a system that models visual and lexicon information in one model using weighted finite-state transducers, but manually add the ground truth words to their lexicon [5]. Mishra et al. use higher order language priors to improve open-vocabulary word recognition performance [9]. In addition, Neumann et al. have demonstrated a real-time end-to-end solution for text detection and open-vocabulary recognition using extremal regions [10], [11].

One of the main differences between the work we present here and these existing solutions is the technique used to detect character locations. Many recent techniques use a sliding window approach to evaluate all possible locations and sizes to find possible characters [5], [9], [12]. These approaches avoid relying on an initial hard segmentation step, but evaluating all sub-windows is expensive, and there is great potential for confusion when non-text areas exhibit character-like features. In contrast, a text segmentation based method can take advantage of coherence across an image. For example, the color characteristics of easier characters can help identify more difficult characters. In this paper, we demonstrate that a segmentation-based approach can outperform sliding-window based approaches for the task of word recognition.

Another difference is the text segmentation technique we present. Many existing methods for scene text cluster colors in the image to produce several possible segmentations, then choose the one that is most likely to be correct [13], [14], [15]. Similarly, Wang et al. [16] extract color information from confident text regions and use it to create segmentations. Mishra et al. [17] also extract foreground and background colors, and use an MRF model in an iterative graph cut framework. The approach we present in this paper is similar to these methods, but we use color clustering as a starting point to fit a regression model for each image. This allows us to segment a larger class of images, since we can model smooth color changes, which often occur in scene text images.

We also incorporate web-based language information in a new way. Donoser et al. use document frequency counts from a query as a source of global language information in word recognition [18]. They query a search engine to acquire the counts. To improve efficiency, we use the data set Web 1T 5-gram released by Google [19] to obtain term frequency information from a crawl of the web.

III. System Design

In this section we describe a word recognition system with three components, as shown in Figure 1. First, we segment each image into foreground and background components. Given the characters from this segmentation, our goal is to find the best word label given appearance and bigram probabilities for the characters and global language information from a web-based lexicon. We could evaluate the probability of every lexicon word based on this information and choose the word with the maximum probability, but since it contains over 13.5 million words this approach is too expensive. Instead, we describe a fast approximation to this approach. We use the Viterbi algorithm to find an initial word label based on just appearance and bigram probabilities, and then we correct any errors in the initial label by evaluating the probability of lexicon words that are within 2-characters of this label given global language information.

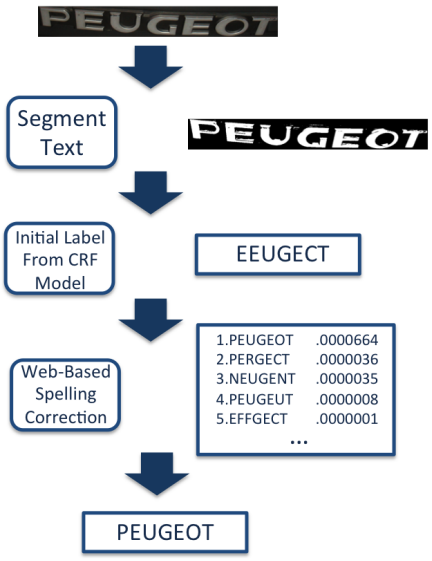


Fig. 1. This describes a step-by-step example of our system. First, an image is segmented into foreground text and background. Next, a conditional random field (CRF) model is used to find the most likely text string, given the connected components in the segmentation. Finally, web-based error correction is performed, where global language and appearance information are combined. The most likely hypothesis is chosen as the final text label.

Below, we begin by presenting our novel text segmentation method. Then, we explain the process for finding an initial word label and describe the fast web-based error correction step in more detail.

A. Text Segmentation

To segment images into foreground text and background, we introduce a technique called bilateral regression segmentation. Scene text images have two characteristics that make them particularly difficult to segment with existing methods. They often contain smooth color changes due to lighting, and are often composed of coherent foreground text and complex backgrounds. To address these challenges, we use a regression model to closely model smooth color changes in images, often allowing for correct segmentation when other methods fail. Additionally, our method only requires modeling the foreground of the image, so complex backgrounds can effectively be ignored.

In this context, the regression

$$z = ax^2 + by^2 + cxy + dx + ey + f$$

represents the quadratic surface that best models the image as a function of pixel location. In order to only model the foreground pixels, we use a weighted regression, where each pixel is weighted according to how close it is to the foreground in feature space. This idea is inspired by the bilateral filtering technique [20], [21]. Pixels that belong to the foreground text will have a high weight while background pixels will have a low weight. This will allow the regression to select out and model the foreground pixels while ignoring the background pixels. We can create a binary segmentation from this model by calculating the amount of error between each pixel and the model and thresholding the error image using a standard

method by Otsu [22]. Pixels that have low error with respect to the model will be part of the foreground text while pixels with high error will be part of the background.

Since we do not know the foreground color a priori, we model the top n colors in each image separately and use a selection procedure to choose the segmentation that is most likely to contain the foreground text. We want that procedure to choose the segmentation with the connected components that can be best recognized as characters. For each connected component in an image, we extract a HOG descriptor, centered over and covering the component. We calculate the l_1 distance to each image in a reference set of synthetic character images from 200 different fonts for 62 character classes [23]. To calculate a score for each image, we take the average of the smallest distance for each component. The image with the lowest score is chosen as the best foreground segmentation.

B. Initial Word Recognition

Given the binary foreground/background image, we use a CRF model to produce an initial text label for each image. We consider each connected component in the binary image as a character and we use a linear-chain CRF to represent the sequence of those characters in a word. Each character can take one of the 62 different labels from the set A-Z, a-z or 0-9. We create appearance features by extracting one HOG descriptor from each character, centered and covering the entire image. These are the same appearance features used in the segmentation step above. We also add a case feature to represent the height of each character. This feature value is the height of a character divided by the height of the tallest character in the same word. We concatenate the HOG descriptor with the case feature value into one feature vector.

We estimate the CRF model parameters with maximum likelihood training by minimizing the negative log-likelihood of the objective function. We use both the ICDAR 2003 Robust Reading training set and the ICDAR 2011 Robust Reading training set as training data. We found that this was not enough data to learn a good model, so we also generated synthetic training data. This was straightforward because we are using binary foreground/background images. We generated our own using a set of synthetic fonts introduced by Weinman et al. [23]. We selected a random word from a dictionary and a random font, and generated each word as white text on a black background. We included words in lowercase, uppercase and title case.

Next, we use the Viterbi decoding algorithm to find an initial word label, given the CRF model [24]. This is a fast, dynamic-programming solution for finding the joint configuration of labels $Y_1, Y_2, ..., Y_n$ that has the highest probability. We also compute three other word labels to encourage case consistency. We know that text is usually written either in all uppercase letters, all lowercase letters, or an uppercase letter followed by all lowercase letters (title case). We compute a word label for each version by restricting the Viterbi algorithm to use only these subsets of characters. Since our model only includes a weak case feature, this method helps to produce labels that follow the case patterns that we expect to see most often. We use the restricted version of the Viterbi model to produce these word labels instead of just transforming the initial word label to have the case patterns since many characters look different in lowercase and uppercase.

C. Web-based Error Correction

We use a web-based error correction step to fix any errors in the initial text labels. First, we construct a lexicon containing the unigrams in the Web 1T 5-gram data set and the frequency count associated with each. The frequency count is the number of times a unigram occurs on web pages. This lexicon contains around 13.5 million words. Since it is created from unigrams that are found on the web, many entries are misspelled words or contain symbols within a word. No preprocessing is performed to remove these errors — the data set contains all unigrams found on the web crawl. Our method is robust to these included entries because we use the frequency count information to favor unigrams that occur more often.

To correct errors, we build a list of hypotheses for possible word labels, evaluate each hypothesis based on the appearance and the global language information obtained from the lexicon, and choose the most likely hypothesis. We begin with the four initial word labels from the previous step, and add hypotheses to this set for all two-character edits of these strings. This means that each hypothesis added must have the same length as the original word labels, but can have up to two characters that are different. We add all two character edits because it allows us to correct a large amount of errors while maintaining a reasonable running time.

Next we calculate the language probability, p_l, for each hypothesis, which is the term frequency count normalized by the sum of all frequency counts in the hypothesis list. To get the final probability of a hypothesis, we multiply this by the appearance probability, p_a, of each character in the word. This value comes from the node marginals from the CRF model trained in the previous step. To summarize, the probability of a hypothesis h with characters $c_1 ... c_n$ in the error correction step is

$$p(h) = p_l(h) * \prod_{i=1}^{n} p_a(c_i).$$

We choose the hypothesis with the highest probability as the final word label for the error correction step. If none of the hypotheses can be found in the lexicon, we back off to the initial word label from the previous step. This allows us to label images with words that are not found in the lexicon.

This error correction step is important because prior to incorporating this global language information, the CRF model used only bigram information. While bigrams are useful for improving labels, they contain local information. In practice, many words contain bigrams that are highly unlikely if looked at alone. For example, the word 'Amherst', contains the characters 'mh', which have a low bigram probability. However, as a word, Amherst is a common town name. To recognize words like this correctly, global language information is required.

IV. Experimental Results

A. Implementation Details

We use a software package for graphical models by Mark Schmidt to implement the CRF model in this paper [25].

Segmentation Method	ICDAR03(FULL)	ICDAR03(50)
Mishra et al. [17]	66.33	74.76
Bilateral Regression	67.76	76.53

Fig. 2. Word accuracy for word spotting on the ICDAR 2003 scene data set of 1107 words.

This package includes standard methods for parameter estimation, inference and decoding. Using this implementation, our method for finding initial word labels is efficient. It took an average of .09 seconds per image to find the four initial word labels.

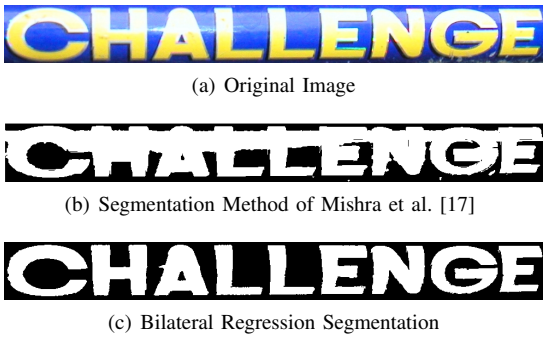
Our techniques for text segmentation and error correction are also efficient. We implemented bilateral regression segmentation and error correction in Matlab and the average running time for our unoptimized segmentation code on a standard desktop is around 3 seconds over the ICDAR 2003 test set. The smallest image in this set is 17 x 12 pixels and the largest is 630 x 1204 pixels. The average size is around 70 x 200 pixels. The average running time for our unoptimized error correction code is also 3 seconds per image.

B. Bilateral Regression Segmentation Evaluation

Since the segmentation of scene text is most often used as an initial step for a recognition process, to evaluate this segmentation technique we compare whether our segmentations allow us to recognize words better than other segmentation methods. We do this by varying the segmentation method used by a complete recognition system. For these experiments we use the word spotting system described in [8]. We use the ICDAR 2003 scene data set, which contains 1107 cropped word images. We use the scene test set instead of the word test set because we were provided segmentations from the state-of-the-art segmentation method published by Mishra et al. [17] for direct comparison. Word spotting also requires a lexicon for each image. We follow the experiments of Wang et al. [2] and use two lexicon sizes. The first contains the ground truth for all images in the data set and is called ICDAR03(FULL). The second contains the ground truth for the image plus 50 random words from the data set and is called ICDAR03(50).

Figure 2 shows the word recognition accuracy for both lexicon sizes for the ICDAR03 scene data set. We compare our technique to the state-of-the-art technique by Mishra et al. [17]. These results show that our segmentation method provides more accurate recognition than existing methods. Additionally, our method is more than an order of magnitude faster than the method by Mishra et al. Their method takes an average of 32 seconds per image while our method takes an average of 3 seconds per image on a standard desktop. Figure 3 shows a sample scene text image with changing colors, the segmentation by the method by Mishra et al., and the segmentation using bilateral regression segmentation. Figure 4 shows more examples of the bilateral regression segmentation for images with complex backgrounds.

We refer readers to the following technical report for additional implementation details and additional experimental results for bilateral regression segmentation [8].



(a) Original Image

(b) Segmentation Method of Mishra et al. [17]

(c) Bilateral Regression Segmentation

Fig. 3. Sample image segmentation using the bilateral regression segmentation technique compared to the state-of-the-art method by Mishra et al. This figure is best viewed in color.

Fig. 4. Sample images with complex backgrounds and their segmentations using bilateral regression segmentation.

	ICDAR 03 (S)	ICDAR 11
Without Error Correction	52.90	41.04
With Error Correction	62.76	48.86

Fig. 5. Word accuracy results with and without web-based error correction.

C. Complete System Evaluation

We evaluate our complete system on the task of open-vocabulary word recognition using two publicly available data sets for scene text recognition. The first data set was created for the ICDAR 2003 Robust Reading competition [6]. It contains 1110 cropped word images with truth labels. In order to compare against existing work, we follow the experiments of Mishra et al. [9] and present results on a subset of this data set. It is created by removing all words with non-alphanumeric characters and all words with less than three characters. The evaluations on this subset are done in a case-insensitive way. The second data set was created for the ICDAR 2011 Robust Reading competition [7]. It contains 1187 cropped word images with truth labels. We present results on the complete data set and, following previous work, evaluate results in a case-sensitive way.

Figure 5 shows the word accuracy of our system with and without error correction. Performance increases by almost 10% on ICDAR 2003 and over 7.5% on ICDAR 2011. This shows the importance of using web-based error correction. Figure 6 shows our results compared to existing methods. On the ICDAR 2003 data set our method increases word accuracy by over 4.5% over the existing state-of-the-art. For the ICDAR 2011 data set, our method increases word accuracy by over 7.5% over the best method submitted to the Robust Reading competition. These results show that our technique outperforms state-of-the-art methods for open-vocabulary word

	ICDAR 03 (S)	ICDAR 11
Neumann's Method [7]	-	33.11
KAIST AIPR System [7]	-	35.60
TH-OCR System [7]	-	41.2
Mishra et al. [9]	57.92	-
Our Method	**62.76**	**48.86**

Fig. 6. Open-vocabulary word accuracy results

Fig. 7. Sample images that we recognize correctly. This image is best viewed in color.

Fig. 8. Sample images that we recognize incorrectly. Characteristics that make these images difficult include low resolution, abrupt lighting changes and low contrast. In addition, words that do not appear in the web-based lexicon, but look similar to something that does can be confused. Here 'lowns' is recognized as 'Towns' and '20p' is recognized as '200'. This image is best viewed in color.

recognition. Figure 7 shows examples of words that were recognized correctly with this system and Figure 8 shows several failure cases.

V. Conclusion

In this paper we present an efficient system for the task of open-vocabulary word recognition. We introduce a novel technique for text segmentation in scene text images. This method improves segmentation for images with smooth color changes due to lighting and images with complex backgrounds. We also demonstrate a new approach to incorporating web-based language information that allows us to take advantage of a lexicon of over 13.5 million words that appear on the web for error correction. In our experiments, we evaluate our text segmentation technique and show that it leads to increased word recognition performance when compared to the current state-of-the-art text segmentation method. In addition, we present state-of-the-art experimental results for open vocabulary word recognition using this complete system on two standard data sets, ICDAR 2003 and ICDAR 2011.

Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. S121000000211. It is also supported by NSF Grant IIS-0916555.

References

[1] K. Wang and S. Belongie, "Word spotting in the wild," in *European Conference on Computer vision*, 2010.

[2] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *International Conference on Computer vision*, 2011.

[3] A. Mishra, K. Alahari, and C. Jawahar, "Top-down and bottom-up cues for scene text recognition," in *Computer Vision and Pattern Recognition*, 2012.

[4] T. Wang, D. Wu, A. Coates, and A. Ng, "End-to-end text recognition with convolutional neural networks," in *International Conference on Pattern Recognition*, 2012.

[5] T. Novikova, O. Barinova, P. Kohli, and V. Lempitsky, "Large-lexicon attribute-consistent text recognition in natural images," in *European Conference on Computer Vision*, 2012.

[6] S. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "Icdar 2003 robust reading competitions," in *International Conference on Document Analysis and Recognition*, 2003.

[7] A. Shahab, F. Shafait, and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images," in *International Conference on Document Analysis and Recognition*, 2011.

[8] J. Feild and E. Learned-Miller, "Scene text recognition with bilateral regression." Department of Computer Science, University of Massachusetts Amherst, Tech. Rep. UM-CS-2012-021, 2012.

[9] A. Mishra, K. Alahari, and C. Jawahar, "Scene text recognition using higher order language priors," in *British Machine Vision Conference*, 2012.

[10] L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," in *Asian Conference on Computer Vision*, 2010.

[11] L. Neumann and J. Matas, "Real time scene text localization and recognition," in *Computer Vision and Pattern Recognition*, 2012.

[12] J. Weinman, "Typographical features for scene text recognition," in *International Conference on Pattern Recognition*, 2010.

[13] C. Thillou and B. Gosselin, "Color binarization for complex camera-based images," in *Proc. Electronic Imaging Conference of the International Society for Optical Imaging*, 2005.

[14] B. Wang, X. Li, F. Liu, and F. Hu, "Color text image binarization based on binary texture analysis," *Pattern Recognition Letters*, vol. 26, 2005.

[15] K. Kita and T. Wakahara, "Binarization of color characters in scene images using k-means clustering and support vector machines," in *International Conference on Pattern Recognition*, 2010.

[16] X. Wang, L. Huang, and C. Liu, "A novel method for embedded text segmentation based on stroke and color," in *International Conference on Document Analysis and Recognition*, 2011.

[17] A. Mishra, K. Alahari, and C. Jawahar, "An mrf model for binarization of natural scene text," in *International Conference onDocument Analysis and Recognition*, 2011.

[18] M. Donoser, H. Bischof, and S. Wagner, "Using web search engines to improve text recognition," in *International Conference on Pattern Recognition*, 2009.

[19] T. Brants and A. Franz, "Web 1t 5-gram version 1," Linguistic Data Consortium, Philadelphia, 2006.

[20] V. Aurich and J. Weule, "Non-linear Gaussian filters performing edge preserving diffusion," in *DAGM Symposium*, 1995.

[21] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *International Conference on Computer Vision*, 1998.

[22] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, 1979.

[23] J. Weinman, E. Learned-Miller, and A. Hanson, "Scene text recognition using similarity and a lexicon with sparse belief propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[24] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, 1967.

[25] M. Schmidt, "UGM: Matlab code for undirected graphical models," http://www.di.ens.fr/ mschmidt/Software/UGM.html, 2013.