Whole is Greater than Sum of Parts: Recognizing Scene Text Words

Vibhor Goel, Anand Mishra, Karteek Alahari, C. V. Jawahar

▶ **To cite this version:**

Vibhor Goel, Anand Mishra, Karteek Alahari, C. V. Jawahar. Whole is Greater than Sum of Parts: Recognizing Scene Text Words. International Conference on Document Analysis and Recognition, Aug 2013, Washington DC, United States. hal-01064766

HAL Id: hal-01064766
https://inria.hal.science/hal-01064766

Submitted on 17 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Whole is Greater than Sum of Parts: Recognizing Scene Text Words

Vibhor Goel[1] Anand Mishra[1] Karteek Alahari[2] C. V. Jawahar[1]

[1]Center for Visual Information Technology, IIIT Hyderabad, India
[2]INRIA - WILLOW / École Normale Supérieure, Paris, France

***Abstract*—Recognizing text in images taken in the wild is a challenging problem that has received great attention in recent years. Previous methods addressed this problem by first detecting individual characters, and then forming them into words. Such approaches often suffer from weak character detections, due to large intra-class variations, even more so than characters from scanned documents. We take a different view of the problem and present a holistic word recognition framework. In this, we first represent the scene text image and synthetic images generated from lexicon words using gradient-based features. We then recognize the text in the image by matching the scene and synthetic image features with our novel weighted Dynamic Time Warping (wDTW) approach.**

We perform experimental analysis on challenging public datasets, such as Street View Text and ICDAR 2003. Our proposed method significantly outperforms our earlier work in Mishra *et al.* (CVPR 2012), as well as many other recent works, such as Novikova *et al.* (ECCV 2012), Wang *et al.* (ICPR 2012), Wang *et al.* (ICCV 2011).

I. Introduction

The document image analysis community has shown a huge interest in the problem of scene text understanding in recent years [6], [15], [19]. This problem involves various sub-tasks, such as text detection, isolated character recognition, word recognition. Due to recent works [5], [8], [13], text detection accuracies have significantly improved. However, the success of methods for recognizing words still leaves a lot to be desired. We aim to address this issue in our work.

The problem of recognizing words has been looked at in two broads contexts – with and without the use of a lexicon [11], [12], [18], [20]. In the case of lexicon-driven word recognition, a list of words is available for every scene text image. The problem of recognizing the word now reduces to that of finding the best match from the list. This is relevant in many applications, such as: (1) recognizing certain text in a grocery store, where a list of grocery items can serve as a lexicon, (2) robotic vision in an indoor/outdoor environment.

Lexicon-driven scene text recognition may appear to be an easy task, but the best methods up until now have only achieved accuracies in the low 70s on this problem. Some of these recent methods can be summarized as follows. In [18], each word in the lexicon is matched to the detected set of character windows, and the one with the highest score is reported as the predicted word. This strongly top-down approach is prone to errors when characters are missed or detected with low confidence. In our earlier work [12], we improved upon on this model by introducing a framework, which uses top-down as well as bottom-up cues. Rather than pre-selecting a set of character detections, we defined a global model that incorporates language priors (top-down) and all potential characters (bottom-up). In [19], Wang *et al.* combined unsupervised feature learning and multi-layer neural networks for scene text detection and recognition. While both these recent methods improved the previous art significantly, they suffer from the following drawbacks: (i) The need for language-specific character training data. (ii) Do not use the entire visual appearance of the word. (iii) Prone to errors due to false or weak character detections.

In this paper, we choose an alternative path and propose a holistic word recognition method for scene text images. We address the problem in a *recognition by retrieval* framework. This is achieved by transforming the lexicon into a collection of synthetic word images, and then posing the recognition task as the problem of retrieving the best match from the lexicon image set. The retrieval framework introduced in our approach is similar in spirit to the influential work of [16] in the area of handwritten and printed word spotting. We, however, differ from their approach as follows. (1) Our matching score is based on a novel feature set, which shows better performance than the profile features in [16]. (2) We formulate the problem of finding the best word match in a maximum likelihood framework and maximize the probability of two features sequences originating from same word class. (3) We propose a robust way to find the match for a word, where k in k-NN is not hand picked, rather dynamically decided based on the randomness of the top retrievals.

Motivation and Overview. The problem of recognizing text (including printed and handwritten text) has been addressed in many ways. Detecting characters and combining them to form a word is a popular approach as mentioned above [12], [18]. Often these methods suffer from weak character detections as shown in Fig. 2(a). An alternative scheme is to learn a model for words [4]. There are also approaches that recognize a word by first binarizing the image, and then finding each connected component [9]. These methods inherently rely on finding a model to represent each character or word. In the context of scene text recognition, this creates the need for a large amount of training data to cover the variations in scene text. Examples of such variations are shown in Fig. 2(b). Our method is designed to overcome these issues.

We begin by generating synthetic images for the words from the lexicon with various fonts and styles. Then, we compute gradient-based features for all these images as well as the scene text (test) image. We then recognize the text in

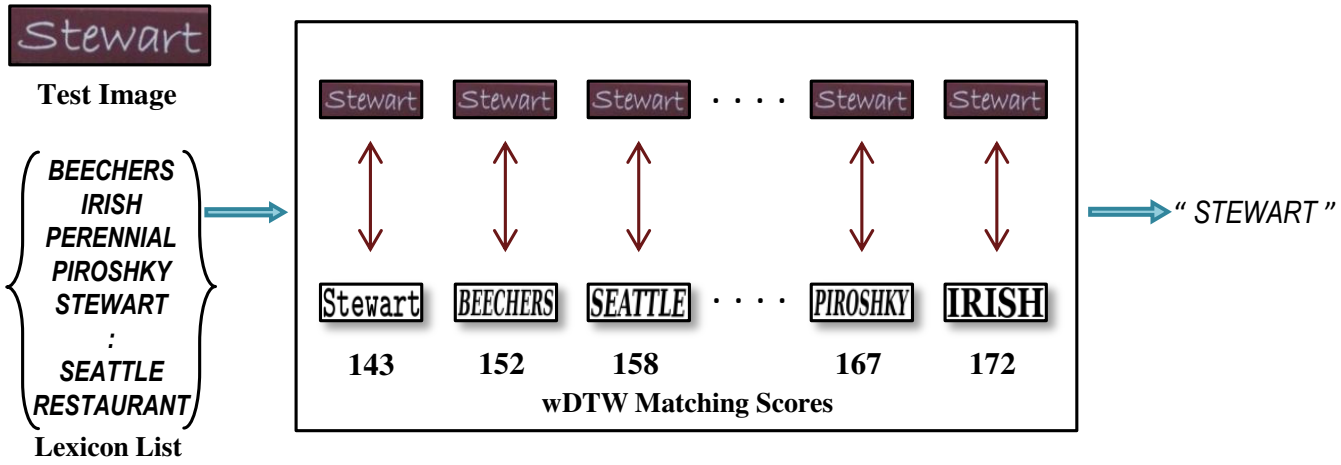


Fig. 1. *Overview of the proposed system. We recognize the word in the test image by matching it with synthetic images corresponding to the lexicon words. A novel gradient based feature set is used to represent words. Matching is done with a weighted DTW scores computed with these features. We use the top k matches to determine the most likely word in the the scene text image.*

the image by matching the scene and synthetic image features with our novel weighted Dynamic Time Warping (DTW). The weights in the DTW matching scores are learned from the synthetic images, and determine the discriminativeness of the features. We use the top k retrieved synthetic images to determine the word most likely to represent the scene text image (see Section II). An overview of our method is shown in Fig. 1.

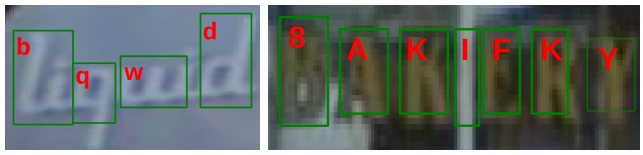
We present results on two challenging public datasets, namely Street View Text (SVT) and ICDAR 2003 (see Section III). We experimentally show that popular features like profile features are not robust enough to deal with challenging scene text images. Our experiments also suggest that the proposed *gradient at edges* based features outperform profile features for the word matching task. In addition to being simple, the proposed method improves the accuracy by more than 5% over recent works [12], [18], [19].

The main contributions of our work are two fold: (i) We show that holistic word recognition for scene text images is possible with high accuracy, and achieve a significant improvement over prior art. (ii) The proposed method does not use any language-specific information, and thus can be easily adapted to any language. Additionally, the robust synthetic word retrieval for scene text queries also shows that our framework can be easily extended for text to image retrieval. However, this is beyond the scope of the paper.

II. Word Representation and Matching

We propose a novel method to recognize the word contained in an image as a whole. We extract features from the image, and match them with those computed for each word in the lexicon. To this end, we present a gradient based feature set, and then a weighted Dynamic Time Warping scheme in the remainder of this section.

Gradient based features. Some of the previous approaches binarize a word image into character vs non-character regions before computing features [9]. While such pre-processing steps can be effective to reduce the dimensionality of the feature space, it comes with its disadvantages. The results of binarization are seldom perfect, contain noise, and this continues to be an unsolved problem in the context of scene text images. Thus, we look for other effective features, which do not rely on binarized images. Inspired by the success of Histogram of Oriented Gradient (HOG) features [7] in many vision tasks, we adapted them to the word recognition problem.



(a) Weak character detections due to high inter-class and intra-class confusion as noted in [12].

(b) Large intra-class variations in scene text words.

Fig. 2. *(a) Character detection is a challenging problem in the context of scene text images. A couple of examples are shown, where weak character detections lead to incorrect word recognition. (b) Large intra-class variations in scene text images makes it challenging to learn models to represent words. Moreover, getting sufficient training data for each word is not trivial.*

To compute the adapted HOG features, we begin by applying the Canny edge operator on the image. Note that we do not expect a clean edge map from this result. We then compute the orientation of gradient at each edge pixel. The gradient orientations are accumulated into histograms over vertical (overlapping) strips extracted from the image. The histograms are weighted by the magnitude of the gradient. An illustration of the feature computation process in shown in Fig. 3. At the end of this step, we have a representation of the image in terms of a set of histograms. In the experimental section we will show that these easy to compute features are robust for the word matching problem.

Matching words. Once words are represented using a set of features, we need a mechanism to match them. The problem

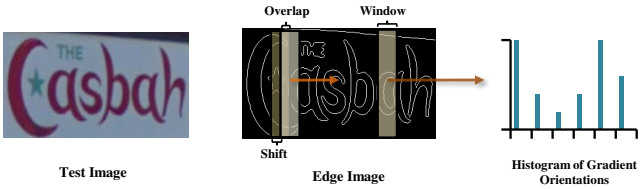


Fig. 3. *An illustration of feature computation. We divide the word image into vertical strips. In each strip we compute histogram of gradient orientation at edges. These features are computed for overlapping vertical strips.*

is how to match the scene text and synthetic lexicon based images[1]. We formulate the problem of matching scene text and synthetic words in a maximum likelihood framework.

Let $X = \{x_1, x_2, \ldots, x_m\}$ and $Y = \{y_1, y_2, \ldots, y_m\}$ be the feature sequences from a given word and its candidate match respectively. Each vector x_i and y_i is a histogram of gradient features extracted from a vertical strip. Let $\omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ represent a set of word images where K is the total number of lexicon words. Since we assume features at each vertical strips are independent, the joint probability that the feature sequences X and Y originate from the same word ω_k, i.e. $P(X, Y|\omega_k)$ can be written as the multiplication of joint probabilities of features originating from the same strip, i.e.,

$$P(X, Y|\omega_k) = \prod_i P(x_i, y_i|\omega_k). \quad (1)$$

In a maximum likelihood framework, the problem of finding an optimal feature sequence Y for a given feature sequence X is equivalent to maximize $\prod_i P(x_i, y_i|\omega_k)$ over all possible Ys. This can be written as minimization of an objective function f, i.e., $\min_Y \sum_i f(x_i, y_i|\omega_k)$. Where f is the weighted squared l^2-distance between feature sequences X and Y i.e.,$f(x_i, y_i) = (x_i - y_j)w_i(x_i - y_j)$. Here w_i is the weight to feature x_i. These weights are learned from the synthetic images, and are proportional to the discriminitiveness of features. In other words, given a feature sequence X and a set of candidate sequences Ys, the problem of finding the optimal matching sequence becomes as minimizing f over all candidate sequences Y. This leads to the problem of alignment of sequences. We propose a weighted dynamic programming based solution to solve this problem. Dynamic Time Warping [17] is used to compute a distance between two time series. The weighted DTW distance $DTW(m, n)$ between the sequences X and Y can be recursively computed using dynamic programming as:

$$DTW(i, j) = \min \begin{cases} DTW(i-1, j) + D(i, j) \\ DTW(i, j-1) + D(i, j) \\ DTW(i-1, j-1) + D(i, j), \end{cases} \quad (2)$$

where $D(i, j)$ is the distance between features x_i and y_j, and the local distance matrix D is written as: $D = (X - Y)^T W(X - Y)$. The diagonal matrix W is learnt from synthetic images. For this we cluster all the feature vectors computed over vertical strips of synthetic images and entropy of each cluster as follows.

[1]Details for generating the synthetic lexicon-based images are given in Section III.

$$H(\text{cluster}_p) = -\sum_{k=1}^{K} Pr(y_j \in \omega_k, y_j \in \text{cluster}_p) \times \log_K(Pr(y_j \in \omega_k, y_j \in \text{cluster}_p)), \quad (3)$$

where Pr is the joint probability of feature y_j originating from class ω_k and falling in cluster$_p$. High entropy of a cluster indicates that the features corresponding to that cluster are almost equally distributed in all the word classes. In other words, such features are less informative, and thus are assigned a low weight during matching. The weight w_j associated with a feature vector y_j is computed as: $w_j = 1 - H(cluster_p)$, if $y_j \in cluster_p$.

Warping path deviation based penalty. To give high penalty to those warping paths which deviate from the near diagonal paths we multiply them with a penalty function $\log_{10}(wp - wp_o)$, where wp and wp_o are warping path of DTW matching and diagonal warping path respectively. This penalizes warping paths where a small portion in one word is matched with a large portion in another word.

Dynamic k-NN. Given a scene text and a ranked list of matched synthetic words (each corresponding to one of the lexicon words), our goal is to find the text label. To do so, we apply k-nearest neighbor. One of the issues with a nearest neighbor approach is finding a good k. This parameter is often set manually. To avoid this, we use dynamic k-NN. We start with an initial value of k and measure the randomness of the top k retrievals. Randomness is maximum when all the top k retrievals are different words, and is minimum (i.e. zero) when all the top k retrieval are same. We increment k by 1 until this randomness decreases. At this point we assign the label of the most frequently occurring synthetic word to a given scene text.

In summary, given a scene text word and a set of lexicon words, we transform each lexicon into a collection of synthetic images, and then represent each image as a sequence of features. We then pose the problem of finding candidate optimal matches for a scene text image in a maximum likelihood framework and solve it using weighted DTW. The weighted DTW scheme provides a set of candidate optimal matches. We then use dynamic k-NN to find the optimal word in a given scene text image.

III. Experiments and Results

In this section we present implementation details of our approach, and its detailed evaluation, and compare it with the best performing methods for this task namely [12], [14], [18], [19].

A. Datasets

For the experimental analysis we used two datasets, namely Street View Text (SVT) [1] and ICDAR 2003 robust word recognition [2]. The SVT dataset contains images taken from Google Street View. We used the SVT-word dataset, which contains 647 images, relevant for the recognition task. A lexicon of 50 words is also provided with each image. The lexicon for the ICDAR dataset was obtained from [18]. Following the protocol of [18], we ignore words with less than two characters or with non-alphanumeric characters, which results in 863 words overall. Note that we could not use the ICDAR 2011 dataset since it has no associated lexicon.

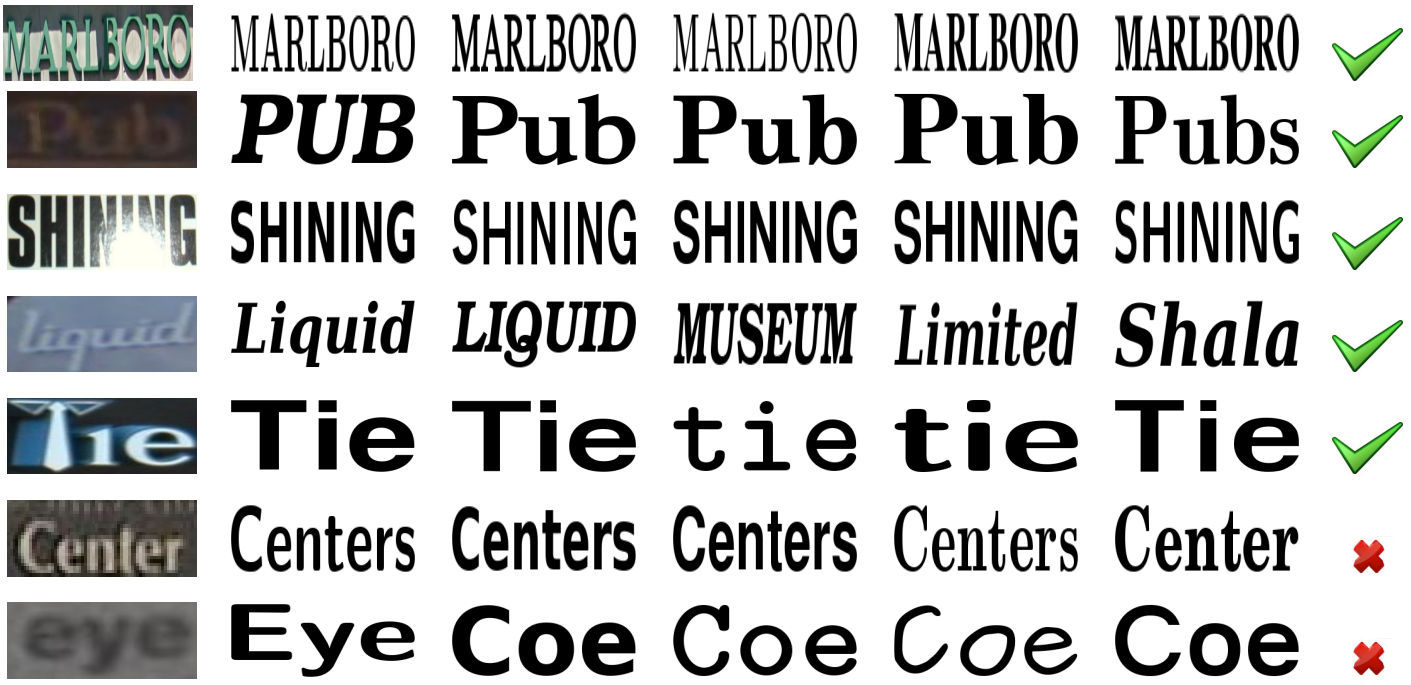


Fig. 4. *Few sample results. Top-5 synthetic word retrieval results for scene text query. First column shows the test image. Top-5 retrieval for the test image are shown from left to right in each row. The icon in the right most column shows whether a word is correctly recognized or not. We observe that the proposed word matching method is robust to variations in fonts and character size. In the fourth row, despite the unseen style of word image "liquid" the top two retrievals are correct. (Note that following the experimental protocol of [18], we do case-insensitive recognition). The last two rows are failure cases of our method, mainly due to near edit distance words (like center and centers) or high degradations in the word image.*

Method	SVT-WORD	ICDAR(50)
Profile features + DTW [16]	38.02	55.39
Gradient based features + wDTW	75.43	87.25
NL + Gradient based features + wDTW	77.28	89.69

TABLE I. *Feature Comparison: We observe that gradient based features outperform profile features for the holistic word recognition task. This is primarily due to the robustness of gradient features in dealing with blur, noise, large intra-class variations. Non-local (NL) means filtering of scene text images further improves recognition performance.*

B. Implementation Details

Synthetic Word Generation. For every lexicon word we generated synthetic words with 20 different styles and fonts using ImageMagic.[2] We chose some of the most commonly occurring fonts, such as Arial, Times, Georgia. Our observations suggest that font selection is not a very crucial step for overall performance of our method. A five pixel-width padding was done for all the images. We noted that all the lexicon words were in uppercase, and that the scene text may contain lowercase letters. To account for these variations, we also generated word images where, (i) only the first character is in upper case; and (ii) all characters are in lower case. This results in $3 \times$ lexicon size $\times$ 20 images in the synthetic database. For the SVT dataset, the synthetic dataset contains around 3000 images.

Preprocessing. Prior to feature computation, we resized all the word images to a width of 300 pixels, with the respective aspect ratio. We then applied the popular non-local means filter smoothing on scene text images. We also remove the stray edges pixels less than 20 in number. Empirically, we did not find this filtering step to be very critical in our approach.

Features. We used vertical strips of width 4 pixels and a 2-pixel horizontal shift to extract the histogram of gradient orientation features. We computed signed gradient orientation in this step. Each vertical strip was represented with a histogram of 9 bins. We evaluated the performance of these features in Table I, in comparison with that of profile features used in [16]. Profile features consist of: (1) projection profile, which counts the number of black pixels in each column. (2) upper and lower profile, which measures the number of background pixels between the word and the word-boundary (3) transition profile, is calculated as the number of text-background transitions per column. We used the binarization method in [10] prior to computing the profile features. Profile features have shown noteworthy performance on tasks such as handwritten and printed word spotting, but fail to cope with the additional complexities in scene text (e.g., low contrast, noise, blur, large intra-class variations). Infact, our results show that gradient features substantially outperform profile based features for scene text recognition.

Weighted Dynamic Time Warping. In our experiments we used 30 clusters to compute the weights. Our analysis comparing various methods are shown in Table I. We observe that with wDTW, we achieve a high recognition accuracy on both the datasets.

Dynamic k-Nearest Neighbor. Given a scene text image to recognize, we retrieve word images from database of synthetic words. The retrieval is ranked based on similarity score. In

[2] www.imagemagick.org/

Fig. 5. *Few images from ICDAR 2003 dataset where our method fails. This may be addressed with inclusion of more variations in our synthetic image database.*

Method	SVT-WORD	ICDAR(50)
ABBYY [3]	35	56
Wang et al. [18]	56	72
Wang et al. [19]	70	**90**
Novikova et al. [14]	72	82
Mishra et al. [12]	73	82
This work	**77.28**	89.69

TABLE II. *Cropped Word Recognition Accuracy (in %): We show a comparison of the proposed method to the popular commercial OCR system ABBYY and many recent methods. We achieve a significant improvement over previous works on SVT and ICDAR.*

other words, synthetic words more similar to the scene text word get a higher rank. We use dynamic k-NN with an initial value of $k = 3$ for all the experiments.

We estimate all the parameters on the train sets of respective datasets. Code for synthetic image generation and feature computation will be made available on our project page.[3]

C. Comparison with Previous Work

We retrieve synthetic word images corresponding to lexicon words and use dynamic k-NN to assign text label to a given scene text image. We compared our method with the most recent previous works related to this task, and also the commercial OCR ABBYY in Table II. From the results, we see that the proposed holistic word matching based scheme outperforms not only our earlier work [12], but also many recent works as [14], [18], [19] on the SVT dataset. On the ICDAR dataset, we perform better than almost all the methods, except [19]. This marginally inferior performance (of about 0.3%) is mainly because our synthetic database fails to model few of the fonts in ICDAR dataset (Fig. 5). These type of fonts are rare in the street view images. A specific preprocessing or more variations in the synthetic dataset may be needed to deal with such fonts. Fig. 4 shows the qualitative performance of the proposed method on sample images. We observe that the proposed method is robust to noise, blur, low contrast and background variations.

In addition to being simple, our method significantly improves the prior art. This gain in accuracy can be attributed to the robustness of our method, which (i) does not rely on character segmentation rather do holistic word recognition; and (ii) learns discriminitiveness of features in a principled way and use this information for robust matching using wDTW.

IV. Conclusion

In summary, we proposed an effective method to recognize scene text. Our method neither requires character segmentation nor relies on binarization, but instead performs holistic word recognition. We show a significantly improved performance over the most recent works from 2011 and 2012. We thus establish a new state-of-the-art on lexicon-driven scene text recognition. The robustness of our word matching approach shows that the natural extension of this work can be in direction of "text to scene image" retrieval. As a part of future work, we would explore the benefits of introducing a hidden Markov models for this problem.

Acknowledgments. This work is partly supported by MCIT, New Delhi. Anand Mishra is supported the Microsoft Research India PhD fellowship 2012 award. Karteek Alahari is partly supported by the Quaero programme funded by the OSEO.

[3] cvit.iiit.ac.in/projects/SceneTextUnderstanding/

References

[1] Street View Text dataset, http://vision.ucsd.edu/~kai/svt/.

[2] Robust word recognition dataset. http://algoval.essex.ac.uk/icdar/RobustWord.html.

[3] ABBYY Finereader 9.0. http://www.abbyy.com/.

[4] J. Almazan, A. Gordo, A. Forns, and E. Valveny. Efficient exemplar word spotting. In *BMVC*, 2012.

[5] X. Chen and A. L. Yuille. Detecting and reading text in natural scenes. In *CVPR*, 2004.

[6] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng. Text detection and character recognition in scene images with unsupervised feature learning. In *ICDAR*, 2011.

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[8] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *CVPR*, 2010.

[9] D. Kumar, M. N. A. Prasad, and A. G. Ramakrishnan. Maps: midline analysis and propagation of segmentation. In *ICVGIP*, 2012.

[10] A. Mishra, K. Alahari, and C. V. Jawahar. An MRF model for binarization of natural scene text. In *ICDAR*, 2011.

[11] A. Mishra, K. Alahari, and C. V. Jawahar. Scene text recognition using higher order langauge priors. In *BMVC*, 2012.

[12] A. Mishra, K. Alahari, and C. V. Jawahar. Top-down and bottom-up cues for scene text recognition. In *CVPR*, 2012.

[13] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *CVPR*, 2012.

[14] T. Novikova, O. Barinova, P. Kohli, and V. S. Lempitsky. Large-lexicon attribute-consistent text recognition in natural images. In *ECCV*, 2012.

[15] T. Q. Phan, P. Shivakumara, B. Su, and C. L. Tan. A gradient vector flow-based method for video character segmentation. In *ICDAR*, 2011.

[16] T. M. Rath and R. Manmatha. Word image matching using dynamic time warping. In *CVPR*, 2003.

[17] D. Sankoff and J. Kruskal. *Time warps, string edits, and macromolecules: the theory and practice of sequence comparison*. Addison-Wesley, 1983.

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

[19] T. Wang, D. Wu, A. Coates, and A. Ng. End-to-end text recognition with convolutional neural networks. In *ICPR*, 2012.

[20] J. J. Weinman, E. G. Learned-Miller, and A. R. Hanson. Scene text recognition using similarity and a lexicon with sparse belief propagation. *PAMI*, 2009.