Scene Text Recognition using Co-occurrence of Histogram of Oriented Gradients

Shangxuan Tian *, Shijian Lu †, Bolan Su * and Chew Lim Tan *
* Department of Computer Science, School of Computing, National University of Singapore
Computing 1, 13 Computing Drive, Singapore 117417
Email: {tians, subolan, tancl}@comp.nus.edu.sg
† Visual Computing Department, Institute for Infocomm Research
1 Fusionopolis Way, #21-01 Connexis, Singapore 138632
Email: slu@i2r.a-star.edu.sg

Abstract—Scene text recognition is a fundamental step in End-to-End applications where traditional optical character recognition (OCR) systems often fail to produce satisfactory results. This paper proposes a technique that uses co-occurrence histogram of oriented gradients (Co-HOG) to recognize the text in scenes. Compared with histogram of oriented gradients (HOG), Co-HOG is a more powerful tool that captures spatial distribution of neighboring orientation pairs instead of just a single gradient orientation. At the same time, it is more efficient compared with HOG and therefore more suitable for real-time applications. The proposed scene text recognition technique is evaluated on ICDAR2003 character dataset and Street View Text (SVT) dataset. Experiments show that the Co-HOG based technique clearly outperforms state-of-the-art techniques that use HOG, Scale Invariant Feature Transform (SIFT), and Maximally Stable Extremal Regions (MSER).

I. INTRODUCTION

Recognition of the text in natural scenes has attracted increasing research attention in recent years due to its crucial importance in scene understanding. It has become a very promising tool in different applications such as unmanned vehicle/robot navigation, living aids for visually impaired persons, content based image retrieval, etc. Though optical character recognition (OCR) of scanned document images has achieved great success, recognition of the scene text by using existing OCR systems still has a large space for improvements due to a number of factors.

First, unlike scanned document texts that usually lie over a "blank" document background with similar color, texture, and controlled lighting, scene texts often have a much more variational background that could have arbitrary color, texture, and lighting conditions as illustrated in Fig. 1. Second, unlike scanned document texts that are usually printed in some widely used text font and text size, the scene text could be captured in arbitrary size and printed in some fancy but infrequently used text fonts as illustrated in Fig. 1. Even worse, the font of the scene text may even change within a single word, for the purpose of special visual effects or attraction of the human attention. Third, unlike scanned document texts that usually have a front-parallel view, scene texts captured from arbitrary viewpoints often suffer from the perspective distortion as illustrated in Fig. 1. All these variations make OCR of scene texts a very challenging task. A robust OCR technique is urgently needed that is tolerant to the variations of scene texts as well as their background.

A number of scene text recognition techniques have been reported which can be classified into two categories. The traditional approach first performs certain preprocessing like binarization, slant correction, perspective rectification before parsing scene texts to the existing OCR engines. Chen et al. [1] performed a variant of Niblack’s adaptive binarization algorithm [2] on the detected text region before feeding to OCR for recognition. An iterative binarization method is proposed in [3] on single character using k-means to produce a set of potential binarized characters and then Support Vector Machines (SVM) is used to measure the degree of character-likeness and the one with maximum character-likeness is selected as the optimal result. Recently the Markov Random Field (MRF) model is adopted for binarization in [4] where an auto-seeding technique is proposed to first determine certain foreground and background pixel seeds and then use MRF to segment text and non-text regions.

Another way is to extract feature descriptors and then use classifiers for scene text recognition. The recent approach
first extracts certain features from gray/color images and then trains classifiers for scene text recognition. This approach is studied extensively in [5] where the scene text recognition performance is evaluated by using different feature descriptors including Shape Contexts, Scale Invariant Feature Transform (SIFT), Geometric Blur, Maximum Response of filters, patch descriptor, etc. in combination with bag-of-words model. But the results are not satisfactory to serve as the basis for word recognition. In [6], the authors address the problem by employing Gabor filters and then building a similarity model to measure the distance between characters in their text recognition framework. Maximally Stable Extremal Regions (MSER) is used in [7] to get a MSER mask and extract orientation features along the MSER boundary. In addition, an unsupervised feature learning system is proposed by [8] using a variant of K-means clustering to first build a dictionary and then map all character images to a new representation using the dictionary.

Recently, the classical HOG feature [9] has also been widely used for scene text recognition. As studied in [10], [11], [12], HOG outperforms almost all the other features due to its robustness to illumination variation and invariance to the local geometric and photometric transformations. However, HOG is just a statistics of gradient orientation in each block which does not capture the spatial relationship of neighboring pixels sufficiently. For example, two image patches having similar HOG features may look very different when their pixel locations are rearranged.

Therefore, we propose to recognize the scene text by using an extension of the HOG, namely, co-occurrence HOG (Co-HOG) [13], that captures gradient orientation of neighboring pixel pairs instead of a single image pixel. Co-HOG divides the image into blocks with no overlap which is more efficient than HOG with overlapped blocks [13]. This is essential in real-time text recognition system. More importantly, relative location and orientation are considered with each neighboring pixel, respectively, which is more precise to describe the character shape. In addition, Co-HOG keeps the advantages of HOG, i.e., the robustness to varying illumination and local geometric transformations. Extensive tests show that Co-HOG outperforms other feature descriptors significantly for scene text recognition.

II. CO-OCCURRENCE OF HISTOGRAM OF ORIENTED GRADIENTS

Co-HOG is extended Histogram of Oriented Gradients. It becomes HOG when the offset is (0, 0) as to be illustrated later in this section. In this section, we first explain the general idea of HOG and then show how to extend it to Co-HOG for the scene text recognition task.

A. Histogram of Oriented Gradients

HOG feature [9] is first proposed to deal with human detection task and later becomes a very popular feature in object detection area. When extracting HOG features, the orientations of gradients are usually quantized into histogram bins and each bin has an orientation range. An image is divided into overlapping blocks and in each block, a histogram of oriented gradients falling into each bin is computed and then normalized to overcome illumination variation. The features from all blocks are then concatenated together to form a feature descriptor of the whole image. Fig. 2 illustrates the extraction process of the classical HOG feature. Due to its robustness to illumination variation and invariance to the local geometric and photometric transformations, many scene text recognition works employ HOG for the recognition of texts in scenes. On the other hand, HOG captures orientation of only isolated pixels, whereas spatial information of neighboring pixels is ignored. Co-HOG instead captures more spatial information and is more powerful in scene text recognition to be discussed in the ensuing subsection.

B. Co-occurrence of Histogram of Oriented Gradients

Co-HOG captures spatial information by counting frequency of co-occurrence of oriented gradients between pixel pairs. Thus relative locations are stored. The relative locations are reflected by the offset between 2 pixels as shown in Fig. 3(a). The yellow pixel in the center is the pixel under study and the neighboring blue ones are pixels with different offsets. Each neighboring pixel in blue color forms an orientation pair with the center yellow pixel and accordingly votes to the co-occurrence matrix as illustrated in Fig. 3(b). Therefore, HOG is just a special case of Co-HOG when the offset is set to (0, 0), i.e., only the pixel under study is counted.

The frequency of co-occurrence of oriented gradients is captured at each offset via co-occurrence matrix as shown in Fig. 3(b). Co-occurrence matrix at a specific offset \((x, y)\) is
Co-occurrence matrix

Equation 2 shows how the weighting of interpolation is employed to vote between two neighboring mechanism based on the gradient magnitude where bi-linear itself can not reflect the difference between strong gradient and is quantized into 9 orientation bins.

For color images, the gradient is computed separately for each color channel and the one with maximum magnitude is used. Gradient magnitude is computed as an in the following three steps.

1) Gradient Magnitude and Orientation Computation: Gradient magnitude is computed as an L2 norm of horizontal and vertical magnitude computed by Sobel filter. For color images, the gradient is computed separately for each color channel and the one with maximum magnitude is used. Gradient orientation ranges between $0^\circ - 180^\circ$ (unsigned gradient) and is quantized into 9 orientation bins.

2) Weighted Voting: The original Co-HOG is computed without weighting as specified in Equation 1 [13], which by itself can not reflect the difference between strong gradient and weak gradient pixels. We propose to add in a weighting mechanism based on the gradient magnitude where bi-linear interpolation is employed to vote between two neighboring orientation bins. Equation 2 shows how the weighting of gradient magnitude and orientation bin is combined and Fig. 4 gives a simple illustration.

$$H(\theta_1, \theta_3) \leftarrow H(\theta_1, \theta_3) + M_1 \left(1 - \frac{\alpha - \theta_1}{\theta_2 - \theta_1}\right) + M_2 \left(1 - \frac{\beta - \theta_3}{\theta_4 - \theta_3}\right)$$

$$H(\theta_1, \theta_4) \leftarrow H(\theta_1, \theta_4) + M_1 \left(1 - \frac{\alpha - \theta_1}{\theta_2 - \theta_1}\right) + M_2 \left(1 - \frac{\beta - \theta_4}{\theta_3 - \theta_4}\right)$$

$$H(\theta_2, \theta_3) \leftarrow H(\theta_2, \theta_3) + M_1 \left(\frac{\alpha - \theta_2}{\theta_1 - \theta_2}\right) + M_2 \left(1 - \frac{\beta - \theta_3}{\theta_4 - \theta_3}\right)$$

$$H(\theta_2, \theta_4) \leftarrow H(\theta_2, \theta_4) + M_1 \left(\frac{\alpha - \theta_2}{\theta_1 - \theta_2}\right) + M_2 \left(1 - \frac{\beta - \theta_4}{\theta_3 - \theta_4}\right)$$

where $H$ is the co-occurrence matrix at a specific offset as defined in Equation 1. $M_1$ is the gradient magnitude at location $(p, q)$ and $\alpha$ is its corresponding gradient orientation. $M_2$ is the gradient magnitude at location $(p + x, q + y)$ with corresponding gradient orientation $\beta$. $\theta_1$ and $\theta_2$ denote the neighboring orientation bin centers of $\alpha$, similar to $\theta_3$ and $\theta_4$.

In the proposed weighting scheme, a pixel with very small gradient value can have a fair large weight if its pair pixel has a large gradient value. To avoid such situations, we do not count pixel pairs when at least one of them has very small gradient value.

3) Feature Vector Construction: The obtained block features are first normalized with L2 normalization method. The Co-HOG feature descriptor of the whole image under study can then be constructed by concatenating all the normalized block features.

III. SCENE TEXT RECOGNITION

Characters in scenes can thus be recognized by training a classifier based on the Co-HOG descriptors as described in the last Section. In our implemented system, a linear SVM classifier is trained using LIBLINEAR [14] which is much faster but with similar performance compared with LIBSVM [15] and SVMLight [16]. We train the SVM classifier by using up to 18500 character images to be discussed in the next Section.

IV. EXPERIMENTAL RESULTS

A. Datasets

We evaluate our methods on ICDAR 2003 [17] and SVT [11] datasets. The ICDAR 2003 character dataset has about 6100 characters for training and 5400 characters for testing. The characters are collected from a wide branches of scenes.
<table>
<thead>
<tr>
<th>Method</th>
<th>ICDAR</th>
<th>SVT</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABBYY FineReader 10 [18]</td>
<td>26.6%</td>
<td>15.4%</td>
</tr>
<tr>
<td>GB+NN [5]</td>
<td>41.0%</td>
<td>-</td>
</tr>
<tr>
<td>HOG+NN [10]</td>
<td>51.5%</td>
<td>-</td>
</tr>
<tr>
<td>NATIVE+FERN [11]</td>
<td>64.0%</td>
<td>-</td>
</tr>
<tr>
<td>MSER [7]</td>
<td>67.0%</td>
<td>-</td>
</tr>
<tr>
<td>HOG+SVM [12]</td>
<td>-</td>
<td>61.9%</td>
</tr>
<tr>
<td>Proposed Co-HOG</td>
<td>79.4%</td>
<td>75.4%</td>
</tr>
<tr>
<td>Proposed Co-HOG (Case Insensitive)</td>
<td>83.6%</td>
<td>80.6%</td>
</tr>
</tbody>
</table>

like book covers, road signs, brand logo and other texts randomly selected from various objects. The text font, size, illumination, color and texture corresponding vary greatly. Take character size for example, the character width ranges from 1 to 589 pixels and the character height ranges from 10 to 898 pixels.

For SVT dataset, only the testing part is annotated in [12] for character recognition which contains about 3796 samples. Compared with the ICDAR 2003, this one is more challenging where most of the characters are cropped from business board and brand names taken from Google Street View. They are usually of fancy fonts, low resolution and often suffer from bad illumination as illustrated in Fig. 1. In addition, we add Char74K dataset [5] for training. Thus the training dataset consists of ICDAR 2003 training dataset and Char74K dataset, which have roughly 18,500 characters in total.

In the experiment, we resize each character to $32 \times 32$ pixels and then divide each into $4 \times 4$ blocks before feature extraction. After we get the Co-HOG features, a linear SVM classifier is trained with LIBLINEAR [14] and evaluated on the testing datasets.

### B. Scene Text Recognition Accuracy

The ICDAR 2003 test dataset altogether has 5430 characters. We exclude those that do not belong to the 62 classes (52 upper and lower case English letters plus 10 digits) thus have 5379 characters left. The trained linear SVM is first tested on this dataset.

Experimental results in Table I show that our proposed method outperforms all previous feature descriptors with an accuracy of 79.4%, while the FineReader 10 [18] gets the worst accuracy with 26.6%, largely due to the fact that the FineReader was designed for document text recognition. The result of GB+NN (41.0%) trained on Char74K dataset, is the one that performs the best as reported in [5]. The character recognition accuracy on ICDAR dataset is not given for the HOG+SVM method as reported in [12]. The method in [8] reports an accuracy of 81.7%. On the other hand, that method uses a huge amount of training data that is not available to the public. More importantly, the testing dataset in [8] only consists of 5198 instead of 5379 characters because it re-crops square character patches from the original dataset and ignores those characters along the boundary which is impossible to fit in a square bounding box.

Text case identification is a very challenging task for scene text recognition because scene texts usually lack specific document layout information. For scene texts, some upper case letters and lower case letters like ‘C’ and ‘c’, ‘S’ and ‘s’, ‘O’ and ‘o’ are extremely difficult to distinguish, even for human beings. In fact, letter cases are often annotated incorrectly in the ground truth dataset. Therefore, we also show case insensitive result which further increases the scene text recognition accuracy of the proposed technique up to 83.6%.

In addition, the accuracy of our proposed method on the SVT dataset is 75.4%, which is only 4% lower than that on the ICDAR. This gap may be further reduced by retraining the SVM using 52 classes because there are no digits in the SVT dataset. The above comparison to some degree shows the superiority of our proposed method which works comparatively well even on very different dataset. Currently

![Successfully recognized characters](image1)

![Wrongly recognized characters](image2)
little character recognition accuracy has been reported on this dataset. The most recent work in [12] achieves an accuracy of 61.9% which is much lower compared with our proposed method. If we ignore letter cases, the accuracy of our proposed method goes up to 80.6%.

Fig. 5 shows some challenging examples that are correctly recognized by our propose method and some failure cases. As Fig. 5a shows, the proposed technique is capable of recognizing many challenging characters in scenes. At the same time, many failed cases as illustrated in Fig. 5b are difficult to read even for humans.

C. Discussion

We compute the confusion matrix on the two datasets and add them together as shown in Fig. 6. As Fig. 6 shows, the mistakes concentrate on those confusing letter cases as we discussed earlier. Besides, two most obvious mistakes are the mis-classification between 'I' and 'l', and between '0' and 'O', which even human beings often fail to differentiate correctly.

Certain recognition failure can be explained by several other factors. For example, some characters are mistakenly annotated in both ICDAR and SVT datasets. Besides, there even exists character of size 1×135 pixels in the ICDAR 2003 dataset because some characters are not cropped carefully. The scene text recognition could be greatly improved without these interfering factors.

V. CONCLUSION AND FUTURE WORK

Character recognition has played a crucial role in text recognition in scene images. We propose to use co-occurrence histogram of oriented gradients (Co-HOG) with weighted voting scheme for scene character recognition. Compared with histogram of oriented gradients (HOG), Co-HOG captures more local spatial information but at the same time keeps the advantage of HOG, i.e., the robustness to illumination variation and invariance to local geometric transformation. The results on both ICDAR 2003 and SVT datasets greatly outperform all the previous feature descriptor based methods. The small accuracy gap between those two very different datasets shows the power of the Co-HOG in capturing the shape information of characters under different scenes.

In the future, we will investigate some global features and combine them with Co-HOG to formulate a more accurate scene character recognition technique.

REFERENCES


