Character Extraction in Web Image for Text Recognition

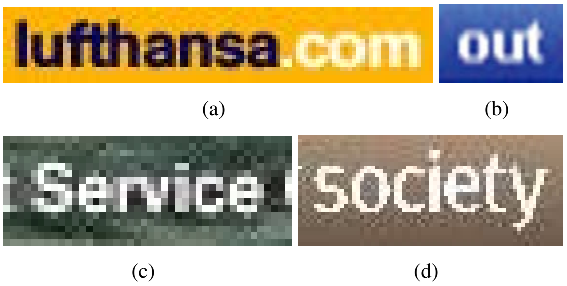
Bolan Su[12*], Shijian Lu[2+], Trung Quy Phan[1] and Chew Lim Tan[1*]
[1]Department of Computer Science,School of Computing,National University of Singapore
Computing 1, 13 Computing Drive, Singapore 117417
[2]Department of Computer Vision and Image Understanding,Institute for Infocomm Research
1 Fusionopolis Way, #21-01 Connexis, Singapore 138632
*{subolan,phanquyt,tancl}@comp.nus.edu.sg,+slu@i2r.a-star.edu.sg

Abstract

Images with text are frequently used on Internet for different purposes. Automatic recognition of text from web images plays an important role on extraction and retrieval of web information. However, the web images are usually in low resolution with artifacts and special effects, which makes word recognition a challenge task even after the text has been localized. In this paper, we propose a robust text recognition technique to efficiently convert the web images into text format. The proposed technique first makes use of the L0 norm smoothing to increase the edge contrast of the input web images. The images are then binarized on each color channel. A connected component analysis is followed to identify the possible character components. Finally the character candidates are recognized by the OCR engine after skew correction. Extensive experiments have been conducted on the latest ICDAR 2011 robust reading competition dataset for born-digital text. The experimental results show the superior performance of our proposed technique.

1. Introduction

The images on Internet are increasing tremendously during these years. Many of these images contain text information that cannot be found in other places of the web pages [2]. The recognition of the textual information within web images is very helpful for a better understanding of the contents of web pages. As these images with text embed are used in Internet for different purposes, text recognition in web images can be applied on different kinds of applications, such as web page indexing & retrieval, web page content filtering [3]. It will become even more important as the textual information within web images is contributing more and more due to the future network development.



(a) (b)

(c) (d)

Figure 1. Some low quality web image examples

Many techniques have been proposed for text extraction and recognition on videos and natural scene images [6, 10], but much fewer efforts have been reported for the recognition of the text within web images [3, 5]. Compared with other images, web images are often more susceptible to certain specific image degradations including low resolution and small size for faster network transmission rate, computer-generate-character artifacts, and special effects on images for attractiveness purpose. As a result, the techniques developed for video/natural scene images often fail to produce satisfactory results when they are directly applied for web images.

The latest Robust Reading Competition in Born-Digital Images (Web and Email) held under the framework of International Conference on Document Analysis and Recognition (ICDAR) 2011 [3] shows current research progress on this area. The contest consists of three tasks, i.e. text localization, text segmentation and word recognition in web image. The third recognition task aims to convert the textual information from bitmap format to ASCII format where the text regions are assumed to have been cropped within web images by certain text localization algorithms. The baseline word recognition method provided by competition organizers achieved around 63% recognition rate on the test dataset. The low recognition rate can largely be attributed to several types of document degradation including low resolution, complex background, low contrast, and non-uniform color as illustrated in Figure 1.

2. Proposed Technique

2.1 Pre-Processing

The input web images need to be resized before applying the smoothing step so that the sizes of characters are large enough for processing. We simply use the bicubic interpolation [4] to resize the input image from (h, w) to (H, W), where (h, w) and (H, W) denote the height and width of the original image and enlarged image, respectively. H is a user defined fixed height to control character size of the input web image, and W is proportional to H, and calculated by $\frac{w}{h} \times H$.

The intensity differences of some web images between text and background can be quite small. Thus the contrast of the input web images need to be stretched for better results. The intensity ranges of the input images are rescaled to $[0, 255]$ at each channel to obtain a significant edge contrast at the text boundaries after smoothing.

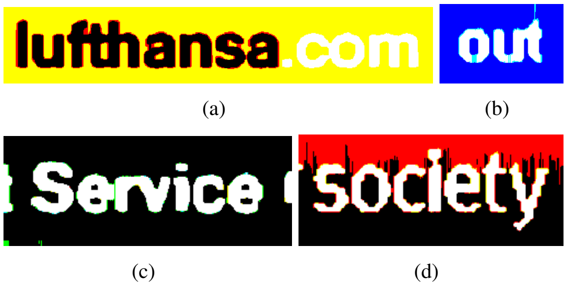
2.2 Image Smoothing and Binarization

To guarantee a good recognition result, the text and background of the web images is better to be smoothed first. In ideal cases, the pixel intensities within the same category (background or text) are equal, intensities differences only occur at the boundary between text and background. Therefore, the smoothed image should have as less intensity change as possible subject to difference with the original line. This problem can be expressed using the following objective function:

$$\underset{S}{min}(\sum_i (S_i - I_i)^2 + \lambda N(S' \neq 0)) \qquad (1)$$

where I denotes the intensities of an original image line, S denotes the intensities of the smoothed image line, i denotes the pixel index, $N(S' \neq 0)$ denotes the number of non zero gradient of S. S' refers to the gradient of S, which is computed by $S_i - S_{i-1}$.

This function is equivalent to the L0-norm Smoothing proposed by Xu et al. [12], and a close-form solution can be found by half quadratic splitting scheme [11] as described in Xu et al.'s paper [12]. Because the gradient between text and background is more important to be evaluated locally [8] and the intensity variation of web images are smaller than that of natural images. If we examine the gradient globally and use the 2-D smoothing manner, some text regions with small intensity differences to the background might be filtered out after 2-D smoothing globally. It is better to apply the smoothing locally on each image line to retain the text information. In the proposed technique, we applied the smoothing procedure two rounds for each web image. The web image is smoothed first row by row, and then column by column at the second round. Smoothed examples can be found in Figure 2.



(a) (b)

(c) (d)

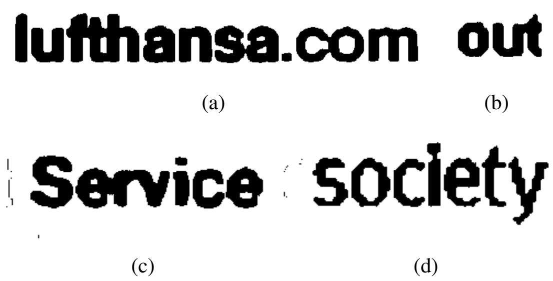
Figure 2. Smoothed images of the original images in Figure 1

2.3 Detection of Character Components

We make use of the shape characteristics of text components to segment text regions from the smoothed images as shown in Figure 1. Since the images are binarized in each channel, the value of each image pixel on each channel is either 0 or 1. There are eight different color levels at most, the color levels vary from $[0, 0, 0]$ to $[1, 1, 1]$. The smoothed images are then segmented into eight fragments, where each fragment contains only image pixels with the same RGB color values. Usually there are less than eight fragments in a web image after segmentation, because most of the web images may contain only little color levels.

Then the character component detection is applied on each fragment separately. Instead of examining the statistics information only for each connected component separately [6], we further consider the whole fragment information since the text pixels are usually at the same fragment. And a few features are used for character region detection as described below.

- Proportion of the pixels in the testing fragment that is also in the boundary components: The boundary



(a) (b) (c) (d)

Figure 3. Binary images of the original images in Figure 1

components refer to the components that touch the image boundary. The text embedded in the web images should locate at the central part of the web images surrounding by background regions. It is obvious that the fragment belongs to background if most of its components locate at the border of the images. If the proportion is larger than a threshold T_{border}, the whole fragment is considered as background regions, otherwise only the components that touch the border are removed.

- Proportion of the pixels in the filled area of all connected components that is also in those component regions: The filled area of a component refers to the area with all the holes filled of the connected component. Since the text strokes are usually strengthened for better visualization, most of those character components of web image fragment do not contain too large hole inside. If the proportion is smaller than a threshold T_{filled}, the whole fragment is considered as non text regions.

- The variation and mean of the heights of connected components within the same fragment: The web images are resized into the same height in the preprocessing step and the characters of the same web images have similar heights, so the characters should have similar sizes. If a fragment is classified as text, the variation of heights of those text components should be smaller than a threshold T_{var}, and average heights of text components should be larger than a fixed height T_{mean}.

The non-character regions of each image fragment are filtered out based on the detection criterion. The remaining character components are then combined together to generate the binary images contain only text. Figure 3 shows the corresponding binary images of those in Figure 1.

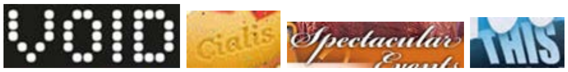


Figure 4. Some web image examples that cannot be recognized

2.4 Skew Correction and Text Recognition

The orientation of the web image text may not be horizontal. The OCR engine may fail to recognize the skew text. The text characters are arranged one by one along the text line, the text pixels are spread out along that direction. So we can use principal component analysis (PCA) [1] to find out the highest variance direction, which should be also the orientation of the text. After the skew correction, the binary web images are fed into the OCR engine to obtain the final text recognition result.

3. Experiments and Discussion

Our proposed technique is tested using the recent Robust Reading Competition for Born-Digital Images dataset [3]. The testing dataset contains 918 web images. Those images have already been extracted from the located text of origin images. Some of these images are easy to recognize, but quite a few of them are challenging. These images usually have various types of problems that decrease its visual quality significantly, such as low resolution, and text artifacts.

During the experiments, we set $T_{boarder} = 0.8$, $T_{filled} = 0.1$, and $T_{var} = \frac{H}{2}, T_{mean} = \frac{H}{6}$, where H is the resized height described in Pre-Processing section. Ground truth texts for the testing dataset are provided by the competition organizers, so we can use Levenshtein distance (edit distance) to compare the difference between the ground truth text and recognized text. We also use recognition rate to evaluate the recognition performance on word level, because in many cases, the textual information is still lost or ambiguous with only part of the texts are recognized. Only the word is recognized completely the same as the ground truth text, the corresponding image will be labeled as successfully recognized.

We compared our results with different methods, including baseline method, recognition after smoothing with Bilateral filters(Bilateral) [9], recognition after binarized with the well-known binarization method(Binarization) [7], and the recognition method for video text(VideoText) [6]. The open source Google

Table 1. Evaluation using Google Tesseract OCR

Methods	Edit Distance	Precision(%)
Baseline	484.5	34.64
Bilateral	468.4	36.06
Binarization	839.8	6.54
VideoText	269.3	55.45
Proposed	317.4	61

Table 2. Evaluation using Abbyy OCR

Methods	Edit Distance	Precision(%)
Baseline	232.8	63.51
Bilateral	228.6	68.63
Binarization	621.1	27.56
VideoText	272.8	61.33
Proposed	**190.1**	**72.33**

Tesseract-OCR and commercial ABBYY OCR are used to obtain the recognition results. The recognition results on the testing dataset of these methods are shown in Tables 1 and 2. The baseline results are obtained by directly applying OCR on the images in testing dataset using Google Tesseract and Abbyy. The results of Abbyy are much better than Google Tesseract, that might be due to some preprocessing work is done by Abbyy before recognition. And the recognition results after binarization are really bad, those classic binarization techniques cannot be applied on web images directly, most of the textual information has lost after binarization. Our proposed method performs best of all the methods. Compared with the bilateral filtering, our proposed smoothing schema works better on edge preserving than bilateral filtering [9], hence more textual information is retained after smoothing.

However, there are still some limitations of our proposed method. Our technique cannot handle those images with text in special fonts, which are shown in Figure 4. We will explore this issue in future study.

4. Conclusion

In this paper, we propose a robust text recognition technique for web images to address the problems described above. The proposed technique that makes uses of the L0-norm smoothing to enhance the contrast between text and background and suppress the intensity variation within text and background of the web images. Experiments have been conducted on the recent Robust Reading Competition dataset held under the latest ICDAR conference [3] to demonstrate that our proposed technique significantly improves the OCR recognition rate.

5. Acknowledgment

This research is supported in part by MOE Grant MOE2011-T2-2-146 (R252-000-480-112)

References

[1] I. Jolliffe. *Principal Component Analysis, Series: Springer Series in Statistics, 2nd Edition*. Springer, 2002.

[2] T. Kanungo and C. Lee. What fraction of images on the web contain text? *International Workshop on Web Document Analysis(WDA)*, pages 43–46, 2001.

[3] D. Karatzas, S. R. Mestre, J. Mas, F. Nourbakhsh, and P. P. Roy. ICDAR 2011 robust reading competition - challenge 1: Reading text in born-digital images (web and email). *International Conference on Document Analysis and Recognition*, pages 1485–1490, September 2011.

[4] R. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Signal Processing, Acoustics, Speech, and Signal Processing*, 29:1153–1160, December 1981.

[5] D. Lopresti and J. Zhou. Locating and recognizing text in WWW images. *Information Retrieval 2*, pages 177–206, 2000.

[6] T. Q. Phan, P. Shivakumara, B. Su, and C. L. Tan. A gradient vector flow-based method for video character segmentation. *International Conference on Document Analysis and Recognition*, pages 1024–1028, September 2011.

[7] J. Sauvola and M. Pietikainen. Adaptive document image binarization. *Pattern Recognition*, 33(2):225–236, 2000.

[8] B. Su, S. Lu, and C. L. Tan. Binarization of historical handwritten document images using local maximum and minimum filter. *International Workshop on Document Analysis Systems*, June 2010.

[9] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *International Conference on Computer Vision(ICCV 1998)*, pages 839–846, 1998.

[10] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. *International Conference on Computer Vision(ICCV 2011)*, 2011.

[11] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, pages 248–272, 2008.

[12] L. Xu, C. Lu, Y. Xu, and J. Jia. Image smoothing via L0 gradient minimization. *ACM Transactions on Graphics (SIGGRAPH Asia 2011)*, 30(6), 2011.