International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.10, (2016), pp.171-178
http://dx.doi.org/10.14257/ijsip.2016.9.10.16

Text Recognition in Mobile Images using Perspective Correction and Text Segmentation

Weisheng Wu, Jian Liu and Lei Li

Modern Educational Technology Center, Pingxiang University, Pingxiang, Jiangxi 337055, P. R. China
190346304@qq.com

Abstract

It is significant that adopt text recognition at mobile devices to care human's health. We observed that although OCR is very suit for recognizing scanned documents, it has poor performation on mobile photoes, which suffer from unequal lighting, clutter, skew, or poor image quality. Therefore, a new algorithm is proposed that take a series of measures to deal with these tough situations of mobile images. This work includes three main steps. Firstly we adopt perspective correction to rectify the distortion of an image. Secondly we use filter to further eliminate the effect of noisy in image. Finally we apply text segmentation to effective measure each text row of image. Compared to OCR text recogniztion success rate 34.7%, the success rate of our method is 65.8%. Experimental results show that the proposed algorithm greatly improves the accuracy of text recognition.

Keywords: *text recognition; OCR; perspective correction; text segmentation*

1. Introduction

It is significant that the spectrum of industries resulting has been applied into document processing, especially from 1950 [1]. Optical Character Recognition (OCR) has converted the scanned documents or image files into completely searchable documents, which can recognize the text content with computers. OCR is widely applied in many fields such as health care industry [2], digital libraries [3], automatic number plate recognition [4], CAPTCHA [5], handwritten recognition [6] and optical music recognition [7].

A classical OCR system includes the following main steps [8]:

(1) Image preprocessing, such as correction of image orientation, perspective correction and noise attenuation;

(2) Image binarization, often carry out adaptively [9];

(3) Segmentation [10], which is frequently hierarchical. First we recognize the page layout and detect text region such as figures, tables *etc.*,. Then it divided into text paragraphs and individual lines. At last these lines are segmented into words, and words are segmented into characters);

(4) Actual recognition, which usually includes two kinds of methods that are unsupervised and unsupervised [11];

(5) postprocessing using the guide of spellchecker.

As the development of mobile devices (mobile phone *etc.*,) having high-quality cameras, using technology of OCR to care the health of people has significant meaning. In this study, a new method was developed to recognize the line items of nutrition facts labels, then it make people easy to record a daily diet log.

We observed that although OCR is very suit for recognizing scanned documents, it has poor performation on mobile photoes, which suffer from unequal lighting, clutter, skew,

ISSN: 2005-4254 IJSIP

or poor image quality [8]. Therefore, we took a series of measures to deal with these tough situations of mobile images. The main contribution of this work includes three advantages. The first one is using perspective correction to rectify the distortion of an image. The second one is adopting filter to eliminate the effect of noisy in image. The last one is using text segmentation to effective measure each text row of image. Experimental results show that the proposed algorithm is effective in the accuracy of word recognition greatly improvement.

2. Image Preprocessing

In order to obtain a high word recognition rate, our image preprocessing addresses two common problems, which are noise removed and perspective distortion. This section includes four steps. First, perspective distortion is corrected using manually operation. Second, the color image is binarized to remove noise. Third, for the noisy bright regions corresponding no line in the fact labels, these regions are filtered out. Finally, each facts line of text is segmented for recognition.

2.1. Perspective Correction

Due to the position of the digital camera is often not orthogonally to the fact table in normal circumstances, the obtained region is a parallelogram or a trapezoid, but not a rectangle that we needed. As shown in Figure 1, an Image is distorted due to improper camera positioning. A large number of experiments demonstrate that these distortions seriously reduce the accuracy of word recognition.

Figure 1. Distorted Image

We first determine the quadrilateral including the fact labels. We can manually carry out this operation or use other effective technique.

We define I as the image including the whole fact labels. We define four points of quadrilateral as following

$$q_{lt} = (x_{q_{lt}}, y_{q_{lt}})$$
$$q_{lb} = (x_{q_{lb}}, y_{q_{lb}})$$
$$q_{rt} = (x_{q_{rt}}, y_{q_{rt}})$$
$$q_{rb} = (x_{q_{rb}}, y_{q_{rb}}) \quad (1)$$

where q_{lt} is left top point, q_{lb} is left bottom point, q_{rt} is right top point, q_{rb} is right bottom point of the quadrilateral.

The new rectangle's dimensions are computed as

$$M_X = \frac{lq_{lt\_rt} + lq_{lb\_rb}}{2}$$
$$M_Y = \frac{lq_{rt\_lb} + lq_{rb\_lt}}{2} \quad (2)$$

where lq_{lt_rt} denotes the length of point qlt to point qrt, as shown in Figure 2.
We define all of the points including fact label in new image as

$$I_M = (i, j), i = 1...M_X, j = 1...M_Y \quad (3)$$

We interpolate the color of the pixel using follow equation

$$q_{lt\_j} = \frac{j \times x_{q_{rt}} + (M_X - j) \times x_{q_{lt}}}{M_X}, q_{lt\_i} = \frac{j \times y_{q_{rt}} + (M_X - j) \times y_{q_{lt}}}{M_X}$$
$$q_{rt\_j} = \frac{j \times x_{q_{rt}} + (M_Y - j) \times x_{q_{lb}}}{M_Y}, q_{rt\_i} = \frac{j \times y_{q_{rt}} + (M_Y - j) \times y_{q_{lb}}}{M_Y}$$
$$q_{lb\_j} = \frac{j \times x_{q_{lb}} + (M_X - j) \times x_{q_{rb}}}{M_X}, q_{lt\_i} = \frac{j \times y_{q_{lb}} + (M_X - j) \times y_{q_{rb}}}{M_X} \quad (4)$$
$$q_{rb\_j} = \frac{j \times x_{q_{lt}} + (M_Y - j) \times x_{q_{rb}}}{M_Y}, q_{rt\_i} = \frac{j \times y_{q_{lt}} + (M_Y - j) \times y_{q_{rb}}}{M_Y}$$

In order to copy point from source image to pixel (i,j) in new image, we use equation (5) to evaluate the point's coordinate (x,y).

Figure 2. Setting the Bounding Rectangle of Text Area

2.2. Binarization

For purpose of region filtering, the color image first is binarized to remove noise. We smooth out the high frequency noise of the image with median filtering, which size is 5*5. The result is shown in Figure 3(a). Next, to decrease intensity of the bright regions and improve shadowed areas, we use adaptive histogram equalization. Using locally adaptive thresholding with patch of 15*15 pixels, image is then binarized, as shown in Figure 3(b). Specifically, we set each pixel in a patch to zero, when the maximum intensity value of this patch is less than 0.35.

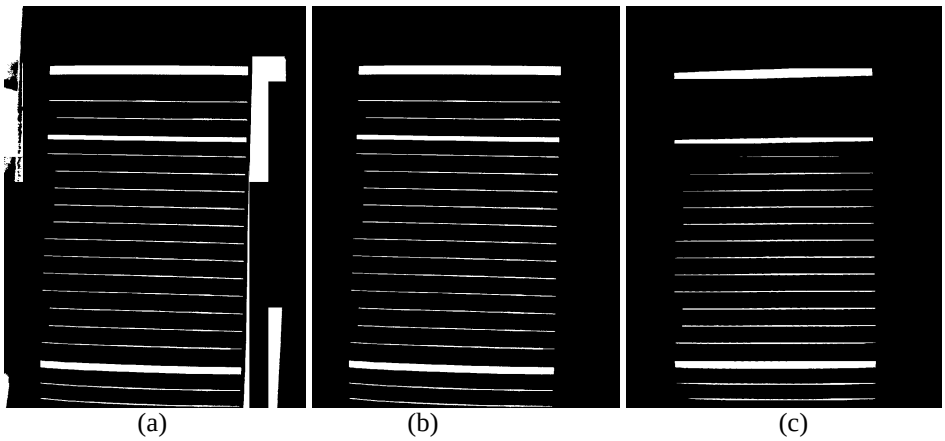
(a) (b)

Figure 3. Image Binarization. (a) Gray Image. (b) Adapt Thresholding Image

2.3. Filtering

After image binarization, the task is eliminating the effect of noisy bright areas, which are not rows in the fact table, as shown in Figure 4(a). We first remove small areas with eccentricity below 0.97, which can make we to obtain thin and long areas as lines. We set the largest areas number equal to 180. According the feature that lines are parallel on fact table, we compute the orientation of each remaining areas and find the largest cluster. We remove the areas that have bigger distance than cluster mean.

We then obtain a fairly clean mask through these filtering steps. In order to correct the image to vertical, we can use a Hough transformation, the result is shown in Figure 4(b). However, there have two common issues to address. One is that some small letters stick on lines because of blur in the source image. The other one is that extra noise to detected lines' left and right. Therefore, to handle these two problems, we adopt extra filtering using column and row strength. We take two effective measures. On the one hand, we set each column containing less white pixels to zero, on the other hand, we set each row including less white pixels to zero. At last, to connect disjoint lines in the mask, we use morphological closing with cell, which structure is thin and long. The result of further filtering is shown in Figure 4(c).



(a) (b) (c)

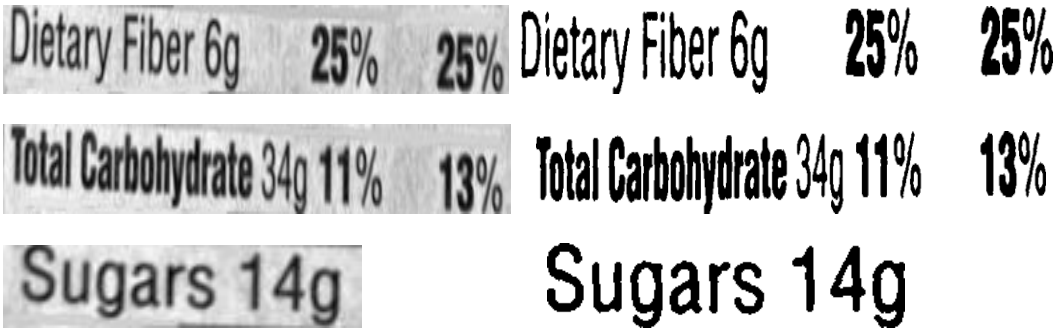
Figure 4. (a) Orientation Filtered. (b) Hough Transformation. (c) Further Filtering

2.4. Text Segmentation

To improve the correct rate of text recognition, we need to take effective measure that segments each text row of image. First, we evaluate text areas' bounding boxes using the

information of lines in the mask. Using the method of Ostu, the corresponding areas are segmented and binarized from the source grayscale image. Second, in general, due to the fact that some small noise areas would be produced by only using binarization, we adopt orientation filtering to further remove noise. At last, we cut every segmented text area through clearing away dark and padding white on the boundary. We then further divide these text areas by the centers of the areas.

Comparison of text segmentation is shown in Figure 5, facts lines are clearly segmented with our method.



(a) (b)

Figure 5. Comparison of Text Segmentation. (a) Direct Text Segmentation (b) Text Segmentation with our Method

3. Recognition

After preprocessed image, our main task is text recognition. The numeric values and text are extracted through optical character recognition. Our OCR engine is a famous open-source engine Tesseract, which is most widely used for text recognition in the world. Nevertheless, the method of is based on character that results in a low accuracy of text recognition, as shown in Figure 6(b). Therefore, to improve the accuracy of text recognition, we use token matching based on word to refine the consequence.

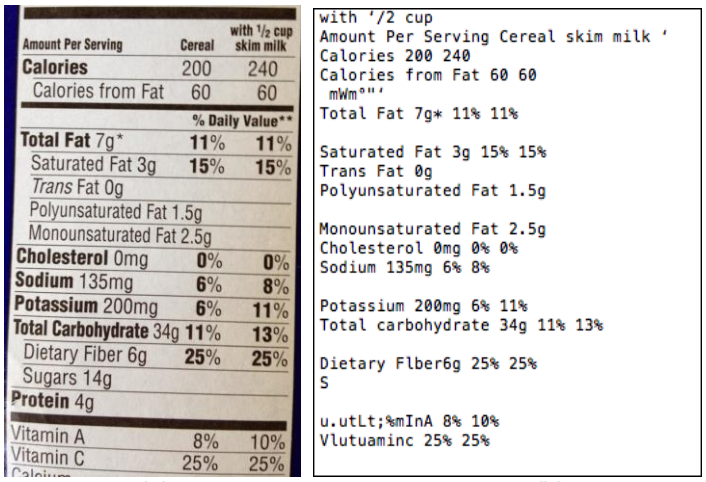
Due to the fact that Tesseract is extremely sensitive to image clutter, skew, and noise, particularly for the unsegmented image, segmenting the image is a useful way to increase the correct rate of text recognition, as shown in Figure 7. The results of the unsegmented image figure and segmented image are compared as shown in Figure 8. It is easy to know that accuracy of text recognition of segmented image is much higher than the value of unsegmented image.

In addition, to enhance our algorithm's capability of correcting misread characters, we limit Tesseract to English alphabet, number, and the special characters such as *, •, and %, as shown in Figure 6(a).

We observe an important thing that the finite number of the words can be counted. Therefore, with predefining a word database containing all of the words of facts label, we can use two token matching algorithms to match Tesseract result to increase our accuracy of text recognition. The first matching algorithm is string matching, which search for precise shapes of tokens in every line. The second matching algorithm is rank matching, which compute the Levenshtein distance between every token in our database and input lines. Normalizing the distance according the length of line, we are able to ensure our algorithm is fair to all tokens of different length. We define that the lower distance has a higher rank and set a threshold to reject obviously inappropriate tokens.

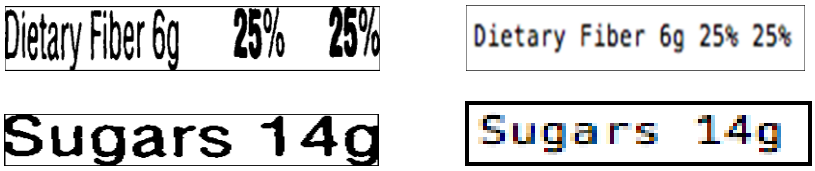
There has a challenge problem that labels compounded of numbers and letters (*e.g.*, 'g', 'mg') are difficult to recognize by directly using string matching or rank matching. The main reason is that the same words do not exist in our word database. Therefore, after obtain both percent and gram values, to eliminate the conflicts between these values, we

adopt the suggested daily values of every facts item. For instance, as shown in Figure 8, display how our method further improves the text recognition.



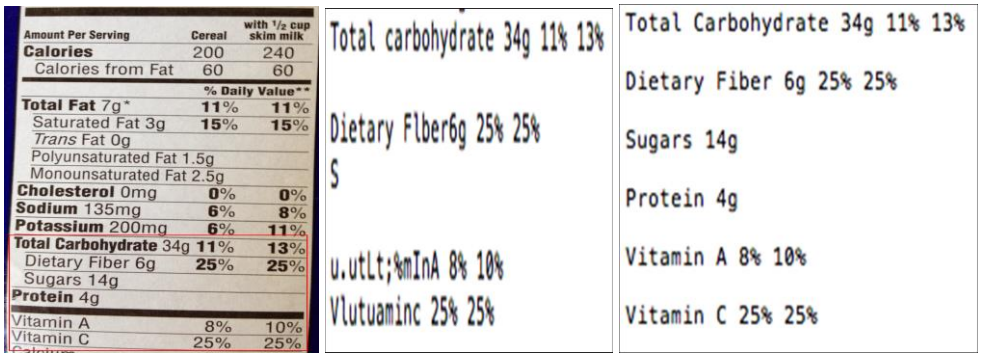
(a) (b)

Figure 6. Text Recognition. (a) Source Image. (b) Result of Text Recognition with OCR



(a) (b)

Figure 7. (a) Segmented Row (b) The Segmented Row Result in Cleaner Text Outputs



(a) (b) (c)

Figure 8. Text Recognition. (a) Source Text Region. (b) Result of Text Recognition with OCR. (c) Result of Text Recognition with word Database

4. Experiments

In order to verify the advantage of our method, we conducted many experiments and compared the results between our algorithm and directly using OCR. We created an image database including 89 images of food items. The values of cholesterol, calories, sodium, fat, dietary fiber, carbohydrates, protein, and sugar on image are manually checked on and used to verify the function of our method.

We obtained a comparison result of our algorithm in error rate and average Time, as shown in Table 1, in different conditions. These conditions included Raw Image (RI),

Naive Match (NM), Segmented Image (SI), Segmented and Perspective Corrected Image (SPCI), and Token Match (TM). Only using raw image and naïve match, the error rate was the highest value 65.3%, which is 11.8% more than using Segmented Image and is 15.5% more than using Segmented and Perspective Corrected Image. Using Segmented and Perspective Corrected Image and token match, by contrast, it got the lowest error rate 34.2%, which is 31.1% less than the biggest error rate. These differences are shown more apparently in figure 9(a). It demonstrates, using image segmentation and perspective correction, our method can yield high quality OCR results.

On the other hand, our algorithm ran on R2013a MATLAB on an Intel dual-core 2.4GHZ with 4GB of RAM. As shown in table 1, the average time was calculated on image database on 89 images. Due to the fact using image segmentation and perspective correction, the average time of our method was 40.5s per image, while the average time of using raw image and naive match was 12.7s per image. These differences are shown more apparently in Figure 9(b). It indicates, using image segmentation and perspective correction, the run time of our method was slower but we obtained high quality result of text recognition.

Table 1. Comparison Result of our Algorithm in Error Rate and Average Time

	RI, NM	SI, NM	SPCI, NM	RI, TM	SI, TM	SPCI, TM
Error rate	65.3%	53.5%	49.8%	61.7%	36.6%	34.2%
Avg Time/s	12.7	34.9	37.3	14.9	37.1	40.5

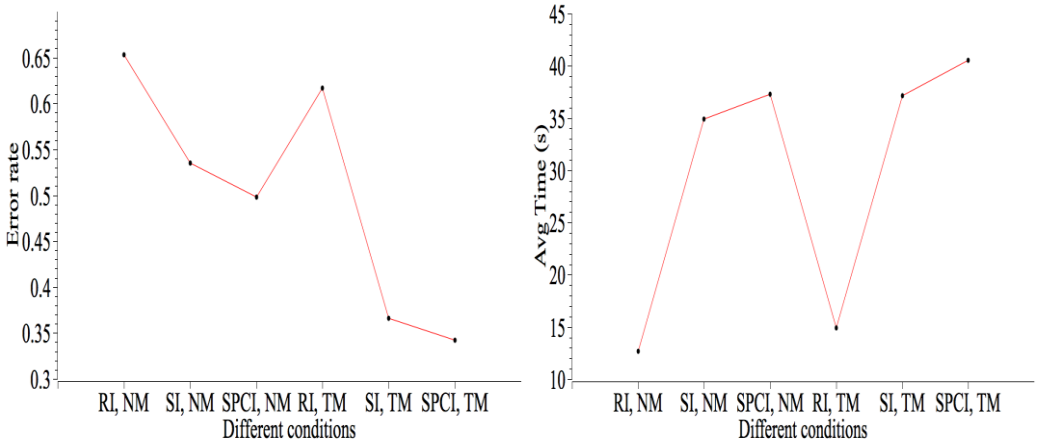


Figure 9. Comparison Result of our Algorithm in Error Rate and Average Time. (a) Error Rate. (b) Average Time

5. Conclusion

This paper presents a new algorithm based on perspective correction and text segmentation, which can effectively improve the accuracy of word recognition. The main contribution of this work includes three advantages. The first one is using perspective correction to rectify the distortion of an image. The second one is adopting filter to eliminate the effect of noisy in image. The last one is using text segmentation to effective measure each text row of image. Experimental results show that the proposed algorithm is effective in the accuracy of word recognition greatly improvement.

However, there are still some challenges for word recognition from image. First, for the bolded characters, the error rate of word recognition is still very high. In the future,

taking advantage of prior knowledge of the word database, we would use template matching to search and erode bolded characters, which can enhance the word recognition. Second, the challange is image blur, which would affect the accuracy of character recognition in some degree. Image blur is difficult to sharpen even using different filters. Third, during image segmentation, as the orientation of high-eccentricity clutter and barcode detection parallel to the facts lines, our text segmentation is hard to discriminate it from table lines.

Acknowledgments

This work was supported by National Natural Science Foundation of China under grant.

References

[1] K. S. Bacchuwar, A. Singh, G. Bansal and S. Tiwari, "An Experimental Evaluation of Preprocessing Parameters for GA Based OCR Segmentation", Proceedings of 2010 The 3rd International Conference on Computational Intelligence and Industrial Application, vol. 2, **(2010)**.
[2] M. D. Ganis, C. L. Wilson and J. L. Blue, "Neural network-based systems for handprint OCR applications", Image Processing, IEEE Transactions on, vol. 7, no. 8, **(1998)**, pp. 1097-1112.
[3] J. Barwick, "Building an institutional repository at Loughborough University: some experiences", Program, vol. 41, no. 2, **(2007)**, pp. 113-123.
[4] S. L. Chang, T. Taiwan, L. S. Chen, Y. C. Chung and S. W. Chen, "Automatic license plate recognition. Intelligent Transportation Systems", IEEE Transactions on, vol. 5, no. 1, **(2004)**, pp. 42-53.
[5] R. Gossweiler, M. Kamvar and S. Baluja, "What's up CAPTCHA? a CAPTCHA based on image orientation", Proceedings of the 18th international conference on World Wide Web, **(2009)**, pp. 84-850.
[6] R. Plamondon and S. N. Srihari, "Online and off-line handwriting recognition: a comprehensive survey", Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 22, no. 1, **(2000)**, pp. 63-84.
[7] A. Singh, K. Bacchuwar, A. Choubey, S. Karanam and D. Kumar, "An OMR based automatic music player", Proceedings of the 3rd International Conference on Computer Research and Development, (**2011**), pp. 174-178.
[8] W. Bieniecki, S. Grabowski and W. Rozenberg, "Image preprocessing for improving OCR accuracy", Proceedings of 2007 International Conference on Perspective Technologies and Methods in MEMS Design, **(2007)**, pp. 75-80.
[9] C. Thillou and B. Gosselin, "Robust thresholding based on wavelets and thinning algorithms for degraded camera images", Proceedings of 2004 International Conference on Advanced Concepts for Intelligent Vision Systems, **(2004)**.
[10] S. Mao, A. Rosenfeld and T. Kanungo, "Document structure analysis algorithms: a literature survey", Proceedings of 2003 Electronic Imaging, **(2003)**, pp. 197-207.
[11] K. Taghva, J. Borsack and A. Condit, "Evaluation of model-based retrieval effectiveness with OCR text", ACM Transactions on Information Systems, vol. 14, no. 1, **(1996)**, pp. 64-93.

Authors

Weisheng Wu, received the BS degree from Jiangxi Normal University in 2003, and the MS degree in computer science from Nanchang University in 2010. Currently, he is a lecturer in Pingxiang University, Pingxiang City, Jiangxi Province, China. His research interests include image processing and software engineering.

Jian Liu, born in 1982, MS. D., his main research interests include image processing and network technique.

Lei Li, born in 1985, BS. D., her main research interest is image processing.

