Nutrition Facts Label Information Extraction

Changxu Luo, Department of Electrical Engineering, Columbia University

***Abstract*—Nutrition facts labels is important for consumers to monitor daily intake, while currently people need to manually take notes on the these data to track the consumption. In this project, a novel system that can extract the structured data from the nutrition labels image is introduced to make it convenient for people to record these labels. The system applies a label area segmentation algorithm, a Tesseract OCR engine, and a label matching algorithm to complete the task. It has a 75% recognition accuracy on 15 common nutrition labels, and an average run time of 7 seconds for a typical phone image[1].**

***Index Terms*—Digital Image Processing, Optical Character Recognition, Image Segmentation, Morphological Operation, Nutrition Label**

I. Introduction

THE US Food and Drug Administration (FDA) requires most packaged food to have nutrition facts label printed on the package. These labels entries, including calories, fat, sodium, carbohydrates, sugars, and proteins, perform as an important reference to instruct consumer's healthy intake[1]. However, consumer may still need to manually take notes on these nutrition values in order to track how much nutrition they have been taken in one day. With the development of mobile devices, current personal devices such as smart phone can take pictures whose quality is high enough to apply digital image processing techniques for label text recognition and give robust output, so that we can use these devices in helping with recording the nutrition information. In this project, a nutrition facts label information extraction procedure is proposed and implemented for extracting structured nutrition label texts from pictures taken by phone cameras. The procedure includes two parts. The first part is the image preprocessing, which includes an algorithm that transform the original image into a comparably high-quality binarized figure with low noise, and segments the figure to extract the the region of the labels. The second part is the optical character recognition (OCR) part, in which the Tesseract OCR engine, as well as a label text matching and correction algorithm, are used to extract the structured label text information from the output image of the first step. The output of the system would be a collection of structured label information for example, $\{$(Calories: 160), (Sugars: 10g)$\}$, etc. The system is tested on the self-collected data, and compare with the manually marked ground truth to measure the accuracy.

II. Image Preprocessing

In this part, the system does two jobs. One is to transform the image and reduce the noise to get a high-quality binarized image, and the other is to apply segmentation algorithm to find the region of the nutrition labels. The entire procedure for this part is shown in figure 1. The implementation applied several basic OpenCV[4] image processing functions. The input image is first sharpened by the following procedure: a mask is calculated as the Gaussian blurring of the initial image with kernel size of 5×5 and $\sigma = 1$, the sharpened image is then calculated as the difference between twice the initial image intensity and the Gaussian mask. The sharpened image is then converted into a gray-scale image, following with a median filter with size of 3 to reduce the noise. After that, the image is downsampled into an image of the same ratio with width of 1000 pixels. Then in order to make the texts clear, a top-hat morphological operation is applied, which is done by minus the result of the image performing an opening operation with the structuring element (SE) from the image. The SE is a rectangle gray-scale element with $1/30$ of the original image size. After this procedure, most of the elements in the figure that is unrelated to the nutrition labels should be removed.

A segmentation algorithm is then applied to the result of the procedure above in order to extract the label area. In order to do this, the feature of the nutrition labels is considered. The standard nutrition labels area has a solid line under each label entry, and normally there is a solid rectangle surrounding the area. Suppose the input image is mainly in the correct orientation, then this feature means that the areas outside the label region should be insensitive to the morphological operations with the SE of horizontal and vertical lines. With this observation, the image first goes into two morphological opening with SE of a horizontal line and a vertical line, whose length are $1/30$ of the image's width and height separately. After that we calculate the difference between these two images, and the result should mainly keeps dark (value 0) in the area outside the label region. Several more morphological operations (closing with vertical and horizontal line SEs) are then applied to the difference image to make sure that the label areas are connected together. And finally, the sizes of all connected components are calculated, and the components whose size are less then 30% of the largest component's size are discarded. The smallest rectangle envelope of the remaining connected components is considered as the segmented region of the nutrition label area. The intermediate result figures of the image preprocessing are shown in figure 2 (a)-(j).

III. Label Recognition

The label recognition performs two tasks. The first task is to apply the OCR algorithm to extract texts from the result image of the preprocessing. The second is to match the texts with the known nutrition label knowledge in order to extract the structured data. In this project, the Tesseract Open Source OCR Engine[3] is applied for the first task. To improve the

[1]Source codes and testing images can be found at https://github.com/camelboat/4830_ELEN_Digital_Image_Processing_Project.

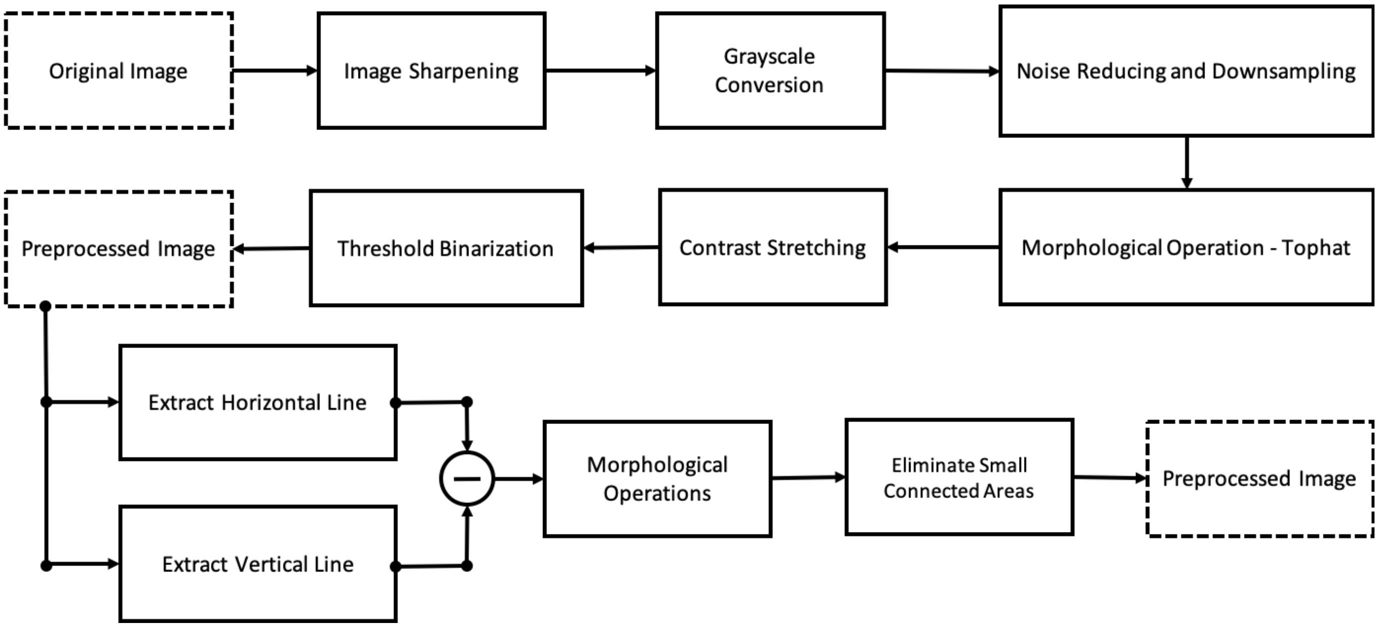

Fig. 1. DAG of image preprocessing algorithm

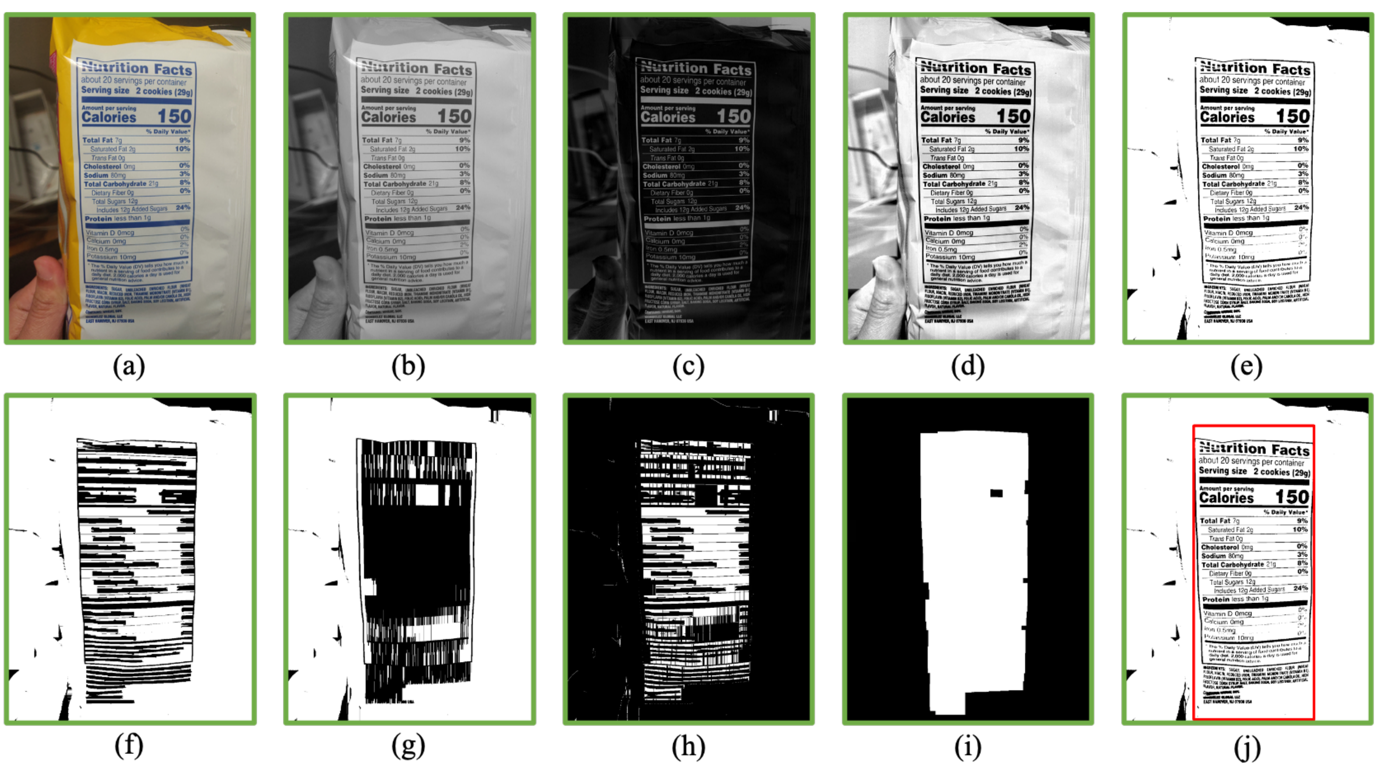

Fig. 2. Intermediate result images of preprocessing. (a): The input RGB image. (b): Gray-scale and downsampled image. (c): Apply morphological Top-hat operation to (b). (d): Apply inverse intensity conversion and contrast stretching to (c). (e): Apply adaptive threshold binarization to (d). (f): Apply opening with horizontal line SE to (e). (g): Apply opening with vertical line SE to (e). (h): Absolute difference image of (f) and (g). (i): Apply morphological closings with horizontal and vertical SEs to (h). (j): Draw the smallest rectangle envelope of several largest connected components in (i) on top of (e), which is considered as the segmented region.

accuracy, the Tesseract configuration is set to detect only the English letters, numbers, space, and characters ".", "%", "*". The output of the Tesseract would be the full texts along with the recognized boxes areas. Since Tesseract assumes that the input image is a scan of the printed article, and the nutrition name on the label is on the same line with its value, one line in the output of Tesseract is considered as one possible nutrition entry. The result of the Tesseract OCR on the example image is shown in figure 3 (a)-(b).

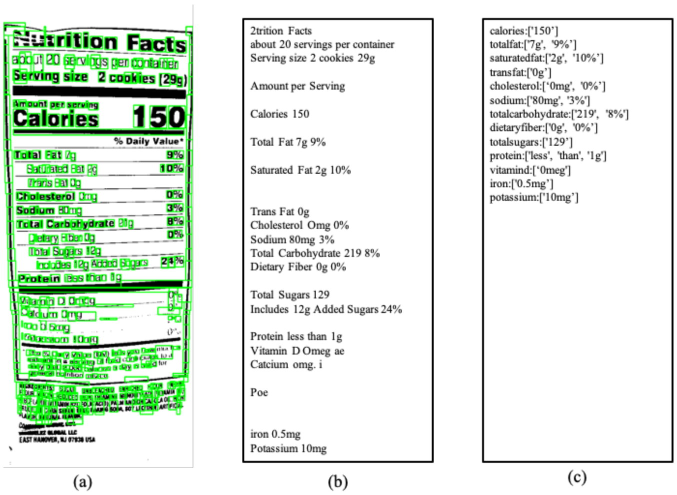

Fig. 3. Result of the Tesseract OCR with proper configuration. (a) Result of recognized boxes areas. (b) Result of recognized texts. (c) Result of structured nutrition data.

When the full text result is got, it is input to a matching algorithm to rectify the error and match the nutrition name. The full text is first split to each line, and each line is then split with the space, and use the first word as the dictionary key. Each key is then compared with the nutrition name list. If the key is not in the list, it is discarded, otherwise, its value would be the rest of words in its line. Then some common errors are recognized to improve the accuracy, such that number "0" at the start of a word could be recognized as letter "O"", and letter "g"" at the end of a word could be recognized as number "9". These errors are recognized through regularized expression matching. If a pattern is matched, and the feature is a valid one instead of being meaningless, that entry would be replaced with the correct feature. For example, the first letter of strings that matches "^(O).*$" is replaced with "0". Each entry in the resulting dictionary is then used to match the established value expression patterns, such as "(*)mg$". The values that are not matched with any of the expression patterns would be discarded. (c) in figure 3 shows the resulting structured data from the example image. The system succeeds in recognizing the entry for "Calories", "Total Fat", "Saturated Fat", "Trans Fat," "Cholesterol", "Sodium", "Total Carbohydrate", "Total Sugars", "Protein", "Vitamin D", "Iron", and "Potassium", which is 13 correct entries out of 15 entries in total.

IV. Experimental Results

The system is tested on 10 different pictures. These pictures are captured through the main camera of OnePlus 6 in default automatic mode. The output result is compared with the manually organized ground truth. 15 common nutrition labels are used as the label dictionary for consideration[2], and the accuracy is calculated as the ratio of the number of corrected recognized labels versus the number of all nutrition labels on the picture that is in the label dictionary. The result is shown in table I. In these 10 testing figures, the system successfully recognized 90 labels among 120 labels, reaching the accuracy of 75%. The average run time is about 7 seconds(Since the image is downsampled to have a width of 1000 pixels, this number doesn't change a lot with the size). All the testing images can be found at the same repository of the project source codes.

TABLE I
Result of the accuray on 10 test images

Picture No.	# Correct Recognized	# Total Labels	Accuracy
1	3	5	60.0 %
2	10	14	71.4 %
3	3	5	60.0 %
4	13	14	92.9 %
5	12	14	85.7 %
6	10	14	71.4 %
7	9	13	69.2 %
8	11	14	78.6 %
9	9	14	64.3 %
10	10	13	76.9 %
Total	90	120	75.0%

V. Discussion, conclusion and Future Work

A. Discussion

In this part, we first show some properties of the figures that have high relation to the recognition accuracy, discuss the reason, and show how to improve on that aspect. First, the picture that has the high recognition accuracy are usually shot under the good lighting condition, with the correct orientation of the label (the rectangle nutrition area aligns well with the image boundary). The lighting issues can be solved through intensity transform in the preprocessing part, and the orientation of the label can be detected in the step of finding the horizontal lines in the image, so that the system can correct the figure orientation by itself.

We also found that the flatness of the area of the nutrition label on the package contributes a lot to the accuracy. This means two types of influences. The first one is a package's soft surface, which is easy to be bumpy. This causes the size of letters on the same line to be different to each other, which makes it difficult to recognize. This should be solved by the OCR algorithm. Another one is the non-flat shape of the package, such as the nutrition labels on a bottle. This makes the line on the label area to be curved on the image and different areas to have difference in focus, bringing problems to both segmentation and word recognition. To solve this problem, perspective transform should be introduced to the preprocessing steps in order to rectify the influence of the package's shape[2].

[2]These nutrition labels include Total Fat, Saturated Fat, Trans Fat, Cholesterol, Sodium, Total Carbohydrate, Dietary Fiber, Total Sugars, Protein, Vitamin D, Calories, Calcium, Iron, and Potassium.

The third problem is that, if the package has intense text regions just besides the nutrition label on the image, the current segmentation algorithm tends to include them into the segmented region, which can bring a little confusion to the label matching step. To solve this problem, the algorithm of finding the nutrition label boundary(suppose that most nutrition label has a solid rectangle line with the same color of the texts inside the region) instead of finding the region, should be developed.

Finally, the algorithm may not perform well when the font of the nutrition labels is not following the FDA standard (font Helvetica). For other irregular fonts, additional training with the OCR program may required for the best recognition performance.

B. Conclusion and Future Work

In this project, a system for extracting the structural information from nutrition facts label image is introduced. The system implemented several preprocessing methods to reduce the noise, and perform nutrition label area segmentation, apply Tesseract OCR algorithm to extract the text information from the preprocessed image, and use a label matching algorithm for generating the structured data. The system achieves a 75% accuracy on recognition of 15 common nutrition labels from pictures taken from phone cameras with an average run time of 7 seconds for a typical phone image. Some future work may include to address the problems announced in the discussion part, and to improve the efficiency to allow the real-time recognition. The system can also be packaged into a mobile application for convenience.

Acknowledgment

This is the report for course project of ELEN 4830 Digital Image Processing at Columbia University. The author would like to thank the course teaching team, including Prof. Christine P. Hendon, Teaching Assistant Ziyi Huang, and Teaching Assistant Yue Feng for their great suggestions and guidance on the project, as well as all the help during the entire semester.

References

[1] J. Aschemann-Witzel, K. G. Grunert, H. C. van Trijp, S. Bialkova, M. M. Raats, C. Hodgkins, G. Wasowicz-Kirylo, and J. Koenigstorfer. Effects of nutrition label format and product assortment on the healthfulness of food choice. *Appetite*, 71:63–74, 2013.

[2] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Pearson, 2018.

[3] Z. P. Ray Smith. Tesseract Open Source OCR Engine Repository. https://github.com/tesseract-ocr/tesseract.

[4] O. D. Team. Opencv Documentation. https://docs.opencv.org/2.4/index.html.