OCR on curved surfaces for extraction of product details

Snehal Padhye
Chester F Carlson Center for imaging science
Rochester Institute Of Technology
sp1471@rit.edu

Abstract

Finding details on any product is time consuming and eventually frustrating. While bar-codes are present on the products, they do not contain details like expiry date, nutrition details etc that affect the decision to purchase the product. Optical character recognition(OCR) is well known in computer vision applications but the varying structure of different products, fonts and the size of the information printed makes it difficult to recognize characters from the products by simple OCR. The aim of this paper to apply techniques on such varying complex surfaces and detect the information that influences the decision to buy the product. A direct application of this would be to have a handy tool to use in marketplace that provides all the required information from the product on taking an image or scanning it. This would make shopping faster and ease the overall experience.

1. Introduction

Often we spend hours in a marketplace choosing the appropriate product from plethora of options. Some factors that influence our choices are nutrition facts, expiry date and price per unit. We spend a good amount of time searching for the written information. There are kiosks installed at some places that read bar-code and display prices for the product on the screen. It would be nice to have all the other required information handy as well. The problem with bar-code is that they do not contain all the information. Especially expiry dates are printed individually on the products. To extract these product information, we can apply the concept of OCR to detect and recognize texts from product images or videos.

Traditionally, OCR applications can be characterized in three ways:

1. application on documents to recognize handwritten texts, digitize texts etc.
2. application in reading number plates on vehicles
3. application in extracting texts from natural scenes.

OCR on documents is a specialized category in which there are few assumptions on the basis of structure of the document and text. The same reason goes with number plate detection; the structure of plate and arrangement of characters are generally fixed. In text detection in natural images, the texts can be on any complex background, with varying colors, geometries and sizes. Keeping the complex attributes in mind, we categorize our application as a subset of text detection from natural images.

The products may have different structures and their images may have different degree of geometrical orientations. This would affect the curvature and resolution of the characters printed on the products. In order to identify the characters correctly on such images, we need to find their geometrical properties as well. The algorithms should be capable of handling all such complexities. Algorithms for OCR can also be broadly classified into three classes[3]:

1. Classic computer vision algorithms.
2. Specialized deep learning approach.
3. Standard deep learning approaches.

Classical computer vision techniques include filtering the image to highlight the areas containing texts, image partition to find text characters candidates, contour detection and segmentation to detect characters, string detection from character grouping and finally text recognition from the detected strings. While these techniques have been successful to give acceptable accuracy, deep learning strategies have emerged to give better accuracy, adaptability and dealing with complex scenarios such as geometric distortion and low resolution. We propose to apply such deep learning technique to apply OCR on varying curved surfaces of the products.

2. Related Work

OCR is prevalent in computer vision research for many years. Problems related to camera captured images such as

Figure 1. Sample images of custom database

perspective distortion due to shot angles, geometrical variations of the surface and uncontrolled illumination has been known from a long time. Many algorithms have been developed in order to mitigate such problem and perform OCR on objects. For geometries of the object, depth measuring instruments have been previously used. Koo et al. [13] used images captured at two different angles to reconstruct the surface of the book for geometrical correction. Like most of the document reconstruction algorithm, they also considered a cylindrical surface model(CSM). Scale Invariant Feature Transform (SIFT) and random sample consensus (RANSAC) are used to find corresponding points on the two images and the parameters of CSM is calculated by optimizing a non linear cost function. Tian et al. [15] used single image to find the geometrical distortion. Gomes et al. [9] used optimum-path forest classifier (OPF), support vector machines (SVM), multi-layer perceptron, k-nearest neighbor(kNN) to detect speed limits from sign boards with an accuracy of 89.19%. Although all the classical techniques gave good performance on printed documents, text detection in natural scenes is still difficult owing to its complexities. Yi et al. [16] used structure based groupings to detect text from natural scenes. In search of a more generalized, accurate and robust technique, deep learning approaches came into the picture. Jaderberg et al.[12] used deep convolutional neural network (CNN) for text recognition . Gupta et al.[10] presented fully convolutional regression network(F-CRN) for text spotting and achieved an accuracy of 98%. Text detection in multi-oriented images [17] is also achieved using fully convolutional networks (F-CNN). Zhou et al. [18] proposed a FCN efficient and accurate scene text detector (EAST) pipeline to detect texts in natural scenes. Bartz et al.[8] proposed semi-supervised end to end scene text detector(SEE), a singe deep neural network to detect and recognize texts in natural scenes. [11] discusses the potentials of faster R-FCN, R-CNN and SSD systems in text detection.

3. Proposed Work

We plan to evaluate different networks [8] [18] [8] and apply the best one to extract product information. The network would basically consist of a feature extractor layer, feature merger layer and output layer. The limitations with existing networks is that they are trained on natural view text dataset which are usually sign boards on streets or banners/advertisement on establishments. Lack of images with various degree of curvatures (like in our case) make these network not necessarily work well with all types of curved surfaces. We plan to evaluate a network that best detects text on any type of product surfaces and fine tune it so that it detects information on various types of product surfaces. We plan to fine tune trained networks with our custom database. The ultimate aim is to work towards a hand held solution for

The (quick) [brown] {fox} jumps! Over the $43,456.78 <lazy> #90 dog & duck/goose, as 12.5% of E-mail from aspammer@website.com is spam. Der „schnelle" braune Fuchs springt über den faulen Hund. Le renard brun «rapide» saute par-dessus le chien paresseux. La volpe marrone rapida salta sopra il cane pigro. El zorro marrón rápido salta sobre el perro perezoso. A raposa marrom rápida salta sobre o cão preguiçoso.	The (quick) [brown] {fox} jumps! Over the $43,456.78 <lazy> #90 dog & duck/goose, as 12.5% of E-mail from aspammer@website.com is spam. Der „schnelle" braune Fuchs springt über den faulen Hund. Le renard brun «rapide» saute par-dessus le chien paresseux. La volpe marrone rapida salta sopra il cane pigro. El zorro marrón rápido salta sobre el perro perezoso. A raposa marrom rápida salta sobre o cão preguiçoso.

Table 1. Text recognition using tesseract

product information extraction. The proposed plan can be divided into following stages:

- Stage 1: Study and evaluate architecture for the information extraction. The questions to be evaluated are: whether to have single network for detection and recognition, whether to have separate networks for detection and recognition, and whether to have a network for detection and use standard networks to recognize letter. Create custom database for product images.

- Stage 2: Evaluate a base network on basis of architecture finalized in stage 1, train on a standard dataset. Fine tune with the custom dataset.

- Stage 3: Evaluate the performance on test samples of custom database. Check how the detection can be improved on curved surfaces.

4. Implementation

4.1. Stage 1

We studied both single and independent network architectures for text detection and recognition. One such way of using single standard architecture was Tesseract [6]. It is a neural net (LSTM) based OCR engine that does text detection as well recognition. One such example is shown in 1. Another network we evaluated [7]included a deep bidirectional LSTM using CNN features as input and trained on Jaderberg et al's synthetic data [4]. We also tried to apply a deep neural network with a CNN stage extracting features fed to a RNN stage pretrained on Synth 90k dataset[1]. However, these methods failed to recognize text from sample product images. The possible reason for it would be characteristics of the data printed on products such that they are dense and non-uniform as compared to text printed in documents. Also, the LSTM network is trained on data that mostly contains sparse sign boards in a natural view. This may also have been a reason for its failure to recognize dense text on a shiny curved surface. We also tried to fine tune with custom database but it did not improve the recognition any further. Owing to these characteristic of text on products, we finalize an approach to use different networks for text detection and recognition.

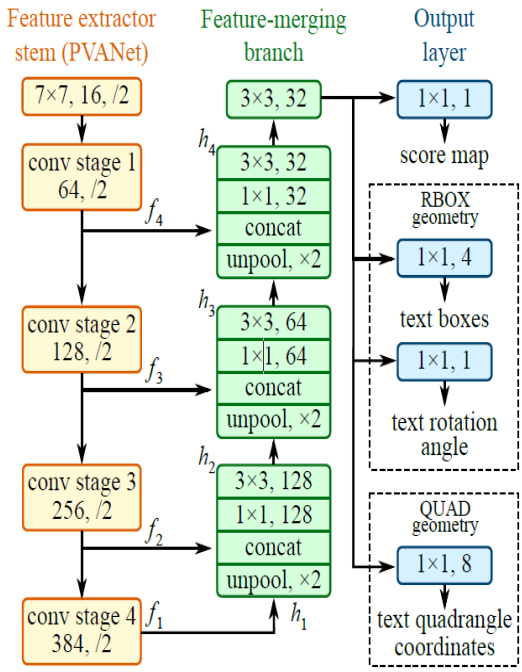


Figure 2. Structure of EAST FCN

4.2. Stage 2

We evaluated standard Single Shot Detector(SSD)[5] to create bounding box on texts as objects. The network could detect the product but could not detect the text separately. We even tried to fine tune this network (pretrained on MS COCO dataset) by giving annotated bounding boxes over region of interest but it still failed to detect individual texts from the product image.

Having assessing various networks, we could understand that we need a robust detection network than a loose detection and recognition network. Another point noted is we could increase our custom database and train and fine tune over more samples. A need for a robust text detection was greatly met by EAST : An efficient and Accurate Scene Detector[18]. It predicts words of arbitrary orientations and quadrilateral shapes in full images with a single neural network.The pre-trained model used [2] achieves 80.83 F1-score on ICDAR 2015 Incidental Scene Text Detection Challenge using only training images from ICDAR 2015 and 2013. The implementation uses ResNet as base network other than PVANET originally used in the paper. The model is a fully-convolutional neural network adapted for text detection that outputs dense per-pixel predictions of words or text lines. This eliminates intermediate steps such as candidate proposal, text region formation and word partition. It's architecture is shown in 2. 2 shows application of the EAST on samples of products. While it gives a very good detection of texts on the product, it misses to

Table 2. Application of EAST on custom samples

Table 3. Application of ASTER on text recognition

accurately box the curved words like 'powder' and 'Oil'.

Since our text detector gives bounding boxes around words and not entire text content, for text recognition, we use Attentional Scene Text Recognizer with Flexible Rectification (ASTER)[14], recognizing cropped text in natural image. It is an end-to-end neural network model that comprises a rectification network and a recognition network. The rectification network adaptively transforms an input image into a new one, rectifying the text in it. Example of text recognition through ASTER is shown in 3. In order to be compatible to our text detector, we take the coordinates predicted by the detector network, extract those patches from original input image and pass it over a loop to ASTER to recognize all the text detected. In order to maintain continuity and meaning of a line, we also sort these coordinates before sending to ASTER to get meaningful text as output.

4.3. Stage 3

4.3.1 Fine Tuning

As we saw in 2, the bounding boxes did not contain all the text at curvatures. In order to correct this detection, we fine

Table 4. Detection result after fine tuning coordinates 1

tuned the network by giving corrected boxes (coordinates) as their labels. We could see slight improvement in the detection for next images 4 5.

4.3.2 Network to learn curvature

We also tried to make the network learn curvature by taking pictures at different angles, stitching them and then giving them as labels for training. The network did not work as expected and it gave bizarre results 6. The main rea-

Table 5. Detection result after fine tuning coordinates 2

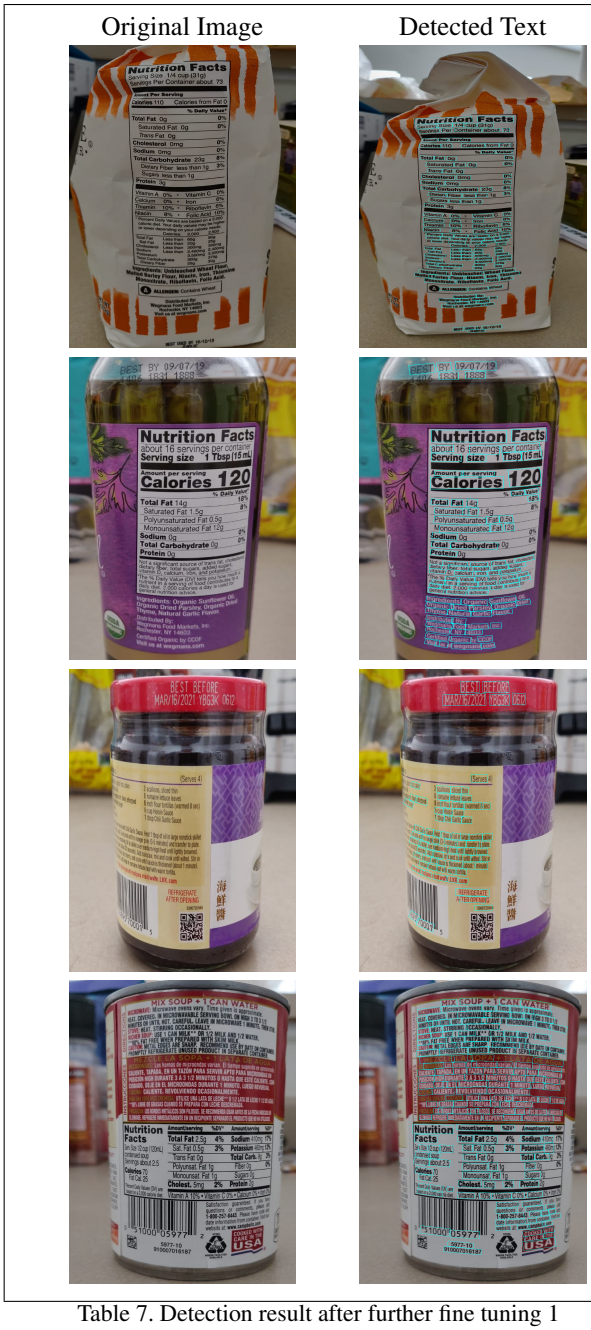


Table 7. Detection result after further fine tuning 1

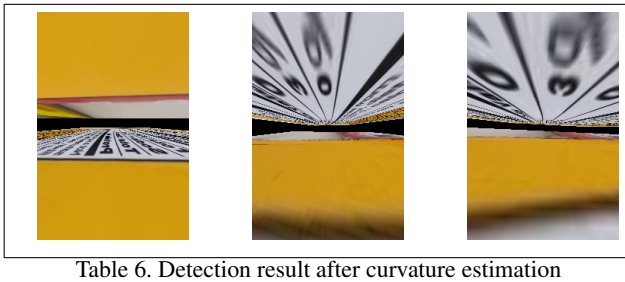


Table 6. Detection result after curvature estimation

son we could speculate about this result would be less number of data to train such model. As there were no standard dataset available for products, we made a custom database 1 of around 80 images which appears to be less for making a network learn a curvature. As we were trying to make a network learn curvature by giving stitched images as input, we thought of testing the algorithm on the stitched image itself so that it has lesser curvatures on text areas and detection could be better. Hence, we tried to give stitched image as input rather than original image and fine tuned it again

Table 8. Detection result after further fine tuning 2

with coordinates. We could see a very slight improvement 78.

5. Result

We combined the results from stage 1,2 and 3 and applied it on our custom test dataset. The final results of text recognition after all above implementation is shown in 9 and 10. The amount and quality of relevant training data played an important role in our experiment. Some standard networks as shown earlier did not work well because of the type/category of data they have trained in. While we could detect text using EAST, we were also limited by amount of data in our custom dataset to make more robust detection.

6. Applications

OCR for product description has many applications. Extraction of nutritional details such as Total fats, sugars, proteins in an eatable product, along with it's expiry date would save a lot of time. The details of the product can be used for comparison of value of the product per unit of consumption. It can also be extended further to extract ingredients to avoid any allergetic reaction.

7. Conclusion

- We tried various networks for text detection and recognition and found separate networks for detection and recognition useful for our use case.
- We used EAST for text detection, fine tuned it for better detection, processed its output to sort detected boxes so that the written information make sense. Lastly, we applied ASTER to recognize this text.
- We could successfully read information on the products but observed that there were not drastic improvements through fine tuning because of the limited data set. There is not much scope in fine tuning bounding boxes because it would go out of the boundaries but not include the curved text. Therefore, the curvature estimation network need to be trained on more number of dataset to improve the detection results.

8. Future Work

- To train curvature estimation with larger dataset.
- Scale it to videos to have text recognition done on scanning.

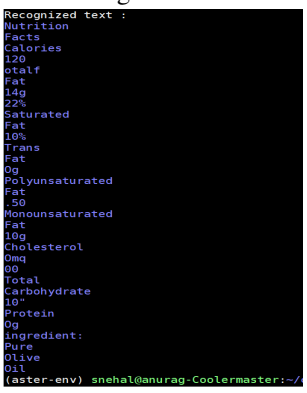
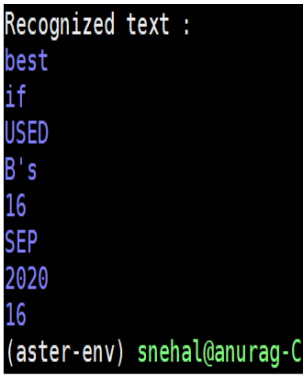
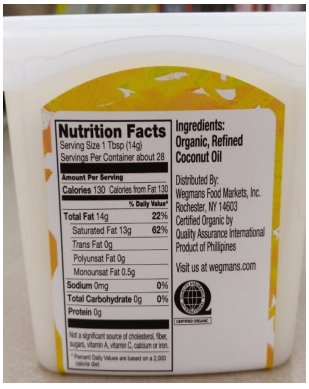
Original Image	Detected Text	Recognized Text
		
		
		
		

Table 9. Final text recognition results 1

Table 10. Final text recognition results 2

References

[1] Crnn. https://github.com/MaybeShewill-CV/CRNN_Tensorflow. Accessed: 2019-03-22.

[2] East. https://github.com/argman/EAST#test. Accessed: 2019-03-22.

[3] A gentle introduction to ocr. https://towardsdatascience.com/a-gentle-introduction-to-ocr-ee1469a201aa. Accessed: 2019-02-22.

[4] Ijcv. https://link.springer.com/article/10.10072Fs11263-015-0823-z. Accessed: 2019-03-22.

[5] Ssd. https://github.com/pierluigiferrari/ssd_keras. Accessed: 2019-03-22.

[6] Tesseract. https://github.com/tesseract-ocr/tesseract. Accessed: 2019-03-22.

[7] Tesseract. https://github.com/weinman/cnn_lstm_ctc_ocr. Accessed: 2019-03-22.

[8] C. Bartz, H. Yang, and C. Meinel. SEE: towards semi-supervised end-to-end scene text recognition. *CoRR*, abs/1712.05404, 2017.

[9] S. L. Gomes, E. d. S. Rebouças, E. C. Neto, J. P. Papa, V. H. C. d. Albuquerque, P. P. Rebouças Filho, and J. M. R. S. Tavares. Embedded real-time speed limit sign recognition using image processing and machine learning techniques. *Neural Computing and Applications*, "28"(1):573–584, Dec 2017.

[10] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2315–2324, June 2016.

[11] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. *CoRR*, abs/1611.10012, 2016.

[12] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1):1–20, 01 2016. Copyright - Springer Science+Business Media New York 2016; Document feature - ; Tables; Equations; Last updated - 2016-02-01; SubjectsTermNotLitGenreText - United States–US.

[13] H. I. Koo, J. Kim, and N. I. Cho. Composition of a dewarped and enhanced document image from two view images. *IEEE Transactions on Image Processing*, 18(7):1551–1562, July 2009.

[14] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2018.

[15] Y. Tian and S. G. Narasimhan. Rectification and 3d reconstruction of curved document images. In *CVPR 2011*, pages 377–384, June 2011.

[16] C. Yi and Y. Tian. Text string detection from natural scenes by structure-based partition and grouping. *IEEE Transactions on Image Processing*, 20(9):2594–2605, Sep. 2011.

[17] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4159–4167, June 2016.

[18] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang. EAST: an efficient and accurate scene text detector. *CoRR*, abs/1704.03155, 2017.