TSB-funded Project 'TADD' - Trainable vision-based anomaly detection and diagnosis Technical Report for February 2014

Ran Song and Tom Duckett

Agri-Food Technology Research Group, Lincoln Centre for Autonomous Systems Research, School of Computer Science, University of Lincoln, UK

Abstract

In our last report, we introduced our method for label detection in a food tray image. The detected label region is essentially a region of interest in which the system will further implement the optical character recognition (OCR). This report will give you a brief background introduction to the typical procedures involved in the OCR process. Also, the challenges that we are facing at the current stage of the project are discussed, which are mainly caused by our specific needs in the integration of automatic label detection and reliable OCR from a complicated image background. Aiming at these challenges, we propose a teaching-by-showing method. It is inspired by the success of our label detection technique based on a similar teaching-by-showing process. Although the preliminary results are promising, we still need to make a great effort in the improvement the proposed method to make it more robust for the real industrial applications.

1. Introduction

We have already developed and demonstrated our label detection technique which can automatically and intelligently detect the label position and orientation. Naturally, the next step is to apply some OCR to read the texts within it. Note that the word 'read' here means the transfer from the appearance and the shapes of the characters to the ASCII codes corresponding to them. The ASCII codes actually form a string in a computing language (e.g., MATLAB or C++) which can be easily used in the direct compari-

son with the correct information (such as date, weight, price, size, etc.). In this way, the system can recognise the tray with a label containing incorrect information.

The rest of the report is organised as follows. In Section 2, we briefly introduce the OCR technique employed in our method. In Section 3, we list the difficulties that need to be conquered to incorporate the OCR into our label-TADD system. In Section 4, we propose our own method to solve the problem and show some preliminary results. Finally, we conclude in Section 5.

2. An overview of the Tesseract OCR

Tesseract is an open-source OCR engine developed by HP. It was released in late 2005 and is now available at http://code.google.com/p/tesseract-ocr. In Tesseract, the first step is a connected component analysis in which outlines of the components are stored. At this stage, outlines are gathered together, purely by nesting, into *Blobs*.

Blobs are then organised into text lines, and the lines and regions are analysed for fixed pitch or proportional text. Text lines are broken into words differently according to the kind of character spacing. Fixed pitch text is chopped immediately by character cells. Proportional text is broken into words using definite spaces and fuzzy spaces.

Next, recognition proceeds as a two-pass process. In the first pass, an attempt is made to recognise each word in turn. Each word that is satisfactory is passed to an adaptive classifier as training data. The adaptive classifier then gets a chance to more accurately recognise text lower down the page. Since the adaptive classifier may have learned something useful too late to make a contribution near the top of the page, a second pass is run over the page, in which words that were not recognised well enough are recognised again.

A final phase resolves fuzzy spaces, and checks alternative hypotheses for the x-height to locate small-cap text.

In Tesseract, since the classifier is able to recognise damaged characters easily, the classifier was not trained on damaged characters. In fact, its classifier was trained on a mere 20 samples of 94 characters from 8 fonts in a single size, but with 4 attributes (normal, bold, italic, bold italic), making a total of 60160 training samples. However, Tesseract contains relatively little linguistic analysis. This could cause potential problems in our applications

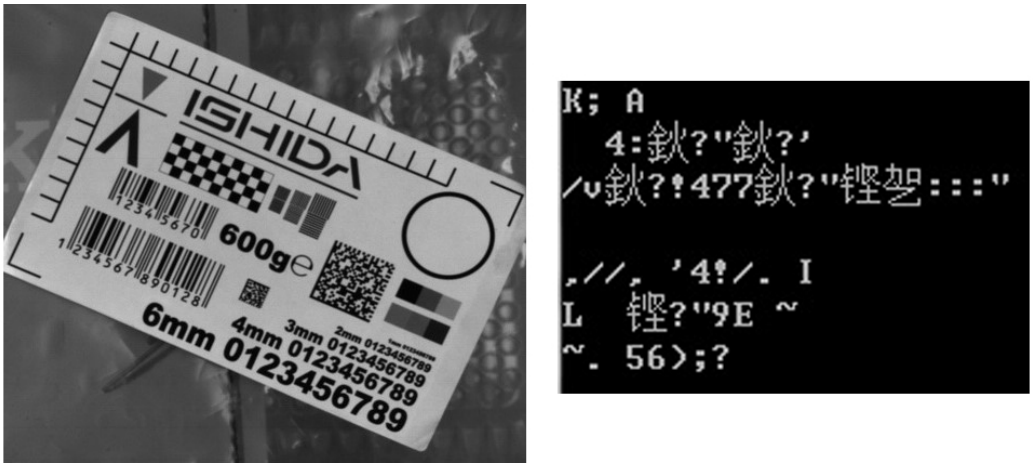

Figure 1: Left: the ROI image output by our label detection algorithm; Right: the results of directly applying Tessearct OCR on it

since the texts on a food tray could be in French, German, Dutch, Welsh, etc. In this project, we do not intend to make any effort on that since it is out of the scope. In the demonstration, we shall only use English texts. The fact is that there is no generic OCR solution due to the variety of languages. For instance, the OCR algorithms for some oriental languages such as the Chinese are quite different from those for Latin languages because the graphical structures of the Chinese characters are significantly different. In practice, we can always use different OCRs to handle text lines with different languages.

3. Challenges

We show Fig. 1 to illustrate the challenges involved in our work. The left image shown in Fig. 1 is the output region-of-interest (ROI) image delivered by our label detection method. Instead of a rotated bounding box, we save the horizontal bounding box of the label region as an image on the disk since an image must be a horizontal rectangle. In this way, we can use the ROI image as the input for the Tesseract OCR. Unfortunately, as shown in the right image, it failed to read the text lines in the label. It is not hard to realise that the failure could be caused by the non-text patterns within the label since they can make the OCR 'confused'. Thus we made a synthetic image as shown in Fig. 2 where we removed the non-text part of the original

Figure 2: Left: the synthetic ROI image; Right: the results of directly applying Tessearct OCR on it

ROI image. And we ended up with the right image of Fig. 2. Certainly, we can observe some improvements but clearly we cannot claim that the OCR is successful here.

The reason behind these failures is that the Tesseract OCR does not have its own page layout analysis, also known as text localisation techniques. Tesseract assumes that its input is a binary image with optional polygonal text regions defined because HP had independently-developed page layout analysis/text localisation technology that was used in products (and therefore not released for open-source). Also, to the best of our knowledge, like most OCR techniques, Tesseract is not very robust to rotation.

Text localisation itself can be a very complicated problem if in particular, there is no learning information. Because of its wide range of applicability, some IT giants such as Google is funding some people to try to solve it in a generic manner (http://cmp.felk.cvut.cz/ neumalu1/). There is no point for us to compete with Google on this trend. Thus, we propose a teaching-by-showing method which focuses specifically on our food tray application.

4. Our method

The teaching-by-showing strategy has been successfully applied for label region detection. Please refer to our previous reports for details.

In the proposed method, first, we need a reference image where the targeted text line is removed and the label in it is roughly horizontal. Fig. 3 shows an example. In practice, such a reference image can be a standalone one (e.g., a printing template) loaded from the disk or detected via our label detection method within a food tray image. Note that in this synthetic image, we deliberately remove the text line at the bottom and use it as the reference image.

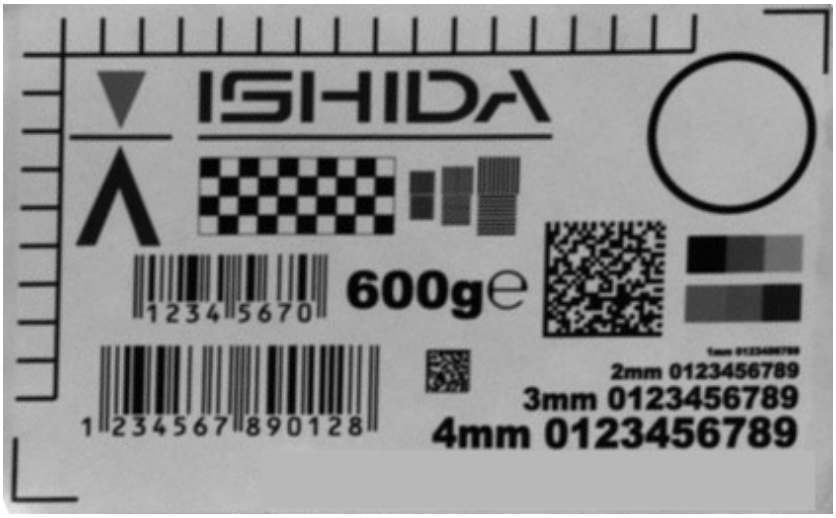

Figure 3: The reference image that we used

We first input this reference image and the real ROI image shown in Fig. 1 and implement the SURF feature detection [1] on both of them. The results are shown in Fig. 4. The following algorithms are similar to our label detection method, including feature matching, RANSAC-based refinement, transform estimation, image warping, textness map generation and text localisation. Therefore, in this report, we do not describe them in detail again. Figs. 4–7 visualise the results of the aforementioned algorithms step by step. Fig. 8 shows that the output of our method—an image saved on the disk which merely contains the one-line text can be easily and correctly recognised by the Tessearct. It is worth pointing out that compared with the previous labet detection method, we make some changes in the textness map generation step. For instance, the erosion and dilation of the difference image is now implemented on a smaller scale.

Figs. 9 and 10 show more results using different test images and reference images.

5. Conclusions

Note that the current version of our text localisation method is based on MATLAB since we just want to check whether our new method can work and work reliably. If it performs really well over a variety of test data, we shall consider further developing it to a full C++ implementation in the

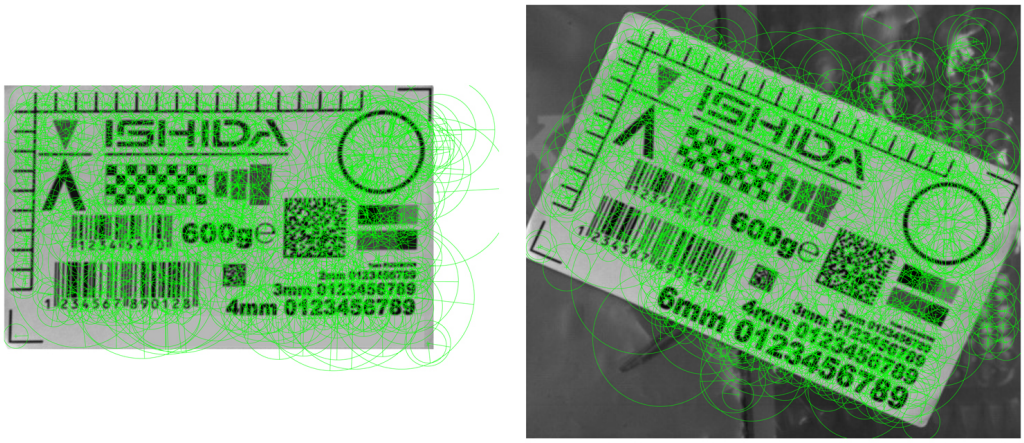

Figure 4: Feature detection on the reference and the ROI images

next stage. Although the results shown in this report is very promising, it is not that difficult to see that it could be not that robust for some labels without enough patterns. Since the proposed method is feature-dependent, its reliability is highly related to the number of features extracted in the region of interest. However, a label on a food tray could be very simple and not textured enough. Hence we still need to further test the current method and see whether we can make some improvement in the near future. In short, efficiency seems not a big problem since the ROI image of a label is typically small; meanwhile, we will concentrate more on the reliability of the method.

References

[1] H. Bay, T. Tuytelaars, L. Van Gool, Surf: Speeded up robust features, in: Proc. ECCV, Springer, 2006, pp. 404–417.

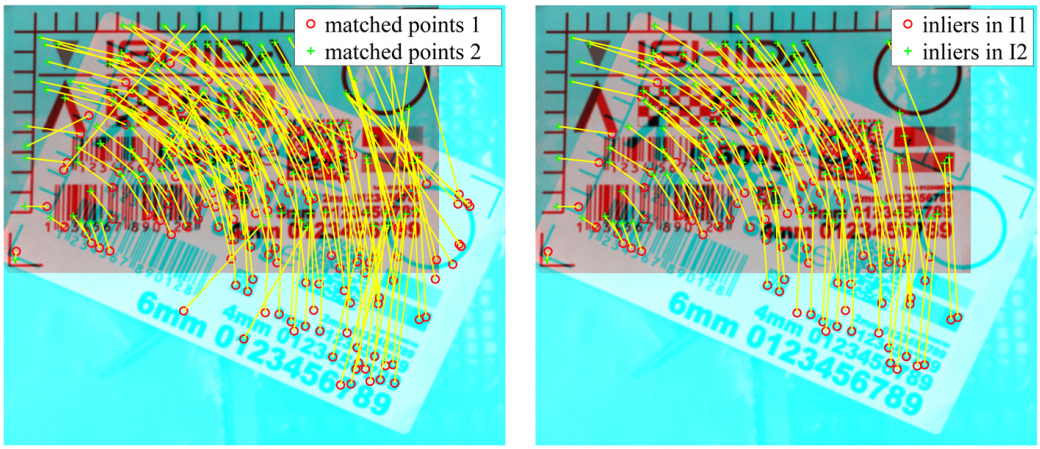

Figure 5: A comparison of the feature matching with and without RANSAC where we shade the two input images in red and blue respectively for a better visualisation result. The yellow lines connect the pairs of matching points which correspond to the same position in the real world. Left: without RANSAC; Right: with RANSAC.

Figure 6: Left: the registered image; Right: the features points in the registered image are now fully overlapped with the ones extracted from the reference image, which means that the registration is very successful.

Figure 7: Left: the 'textness' map where warm colour denotes high textness; Right: the final output of our method is a correct detection of the one-line text that we are interested in. And we save the red rectangle as a digital image on the disk.

Figure 8: The one-line text is correctly recognised by Tessearct.

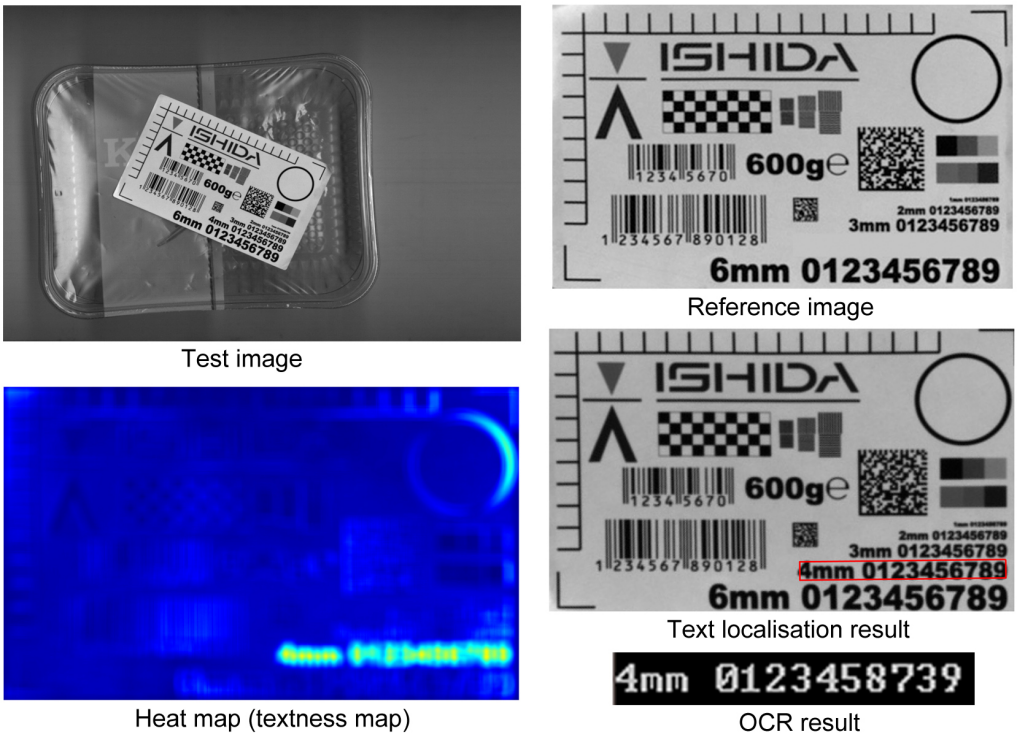

Figure 9: In this test, we used a reference image where the $4mm$ text line in the label was removed.

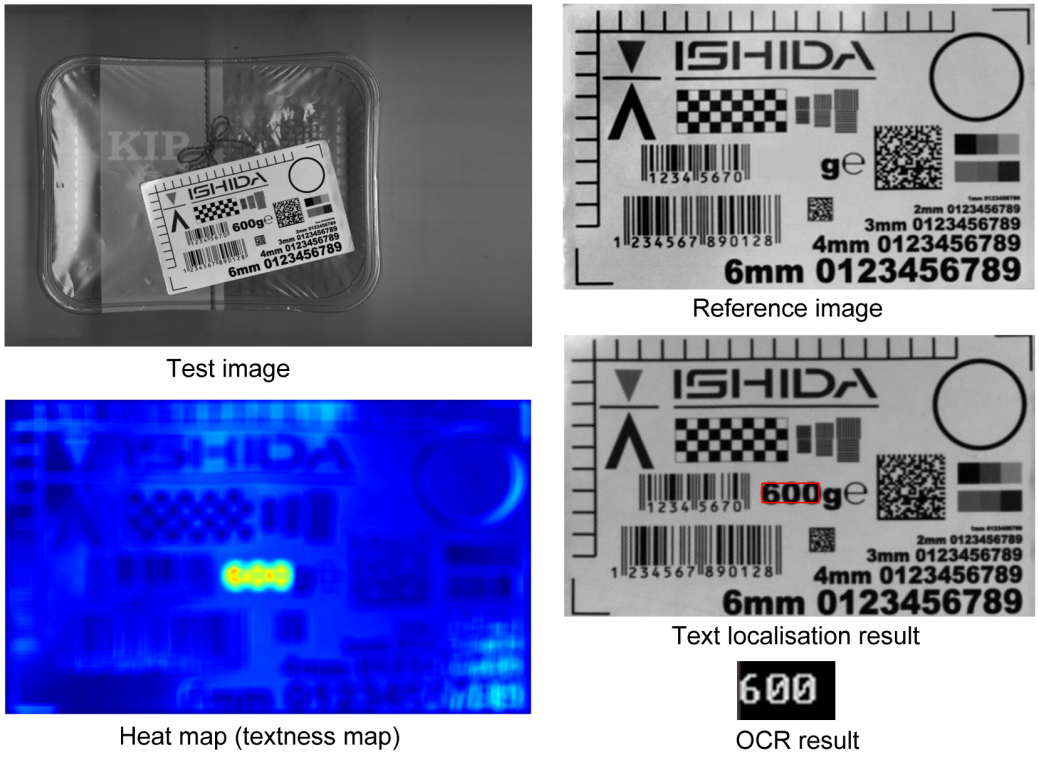

Figure 10: In this test, we used a reference image where the digits of 600 in central region of the label was removed.