DEPARTMENT OF COMPUTER SYATEMS ENGINEERING,
Mehran University of Engineering & Technology, Jamshoro

Menu Translator

Android app for Restaurant Menu card translation from text to image using OCR

Jahanzeb Jabbar (13CS29)
Iqra Urooj (13CS173)
Tahira Ashfaque (13CS185)
Sheeza Rajput (13CS81)
Sundus Abro (13CS13)

ABSTRACT:

—We have developed the android application (MENU TRANSLATOR) that capture the image of menu card and crop the particular dish name ,read the text using OCR and show the image of that dish, and for further details it navigate to the internet ,if internet access is available.

People can use Multilanguage OCR and get dishes according to their countries

By using these scenarios, presenting the pictures of dishes instead of presenting the names of the dishes might be more helpful for people to make good decisions on what they would like to order.

(I) INTRODUCTION:

One of the biggest challenges when traveling is the language barrier. This is especially a big problem when ordering dishes in a restaurant. Because the uniqueness of the names of the dishes, they usually have specific meanings and correspond to distinct dishes, which are hard to imagine by only reading the text on menu. Even though people may have some basic understanding of the literal meaning of these names, the dishes can be totally different from what they think of due to cultural differences. In these scenarios, presenting the pictures of dishes instead of presenting the names of the dishes might be more helpful for people to make good decisions on what they would like to order.

Inspired by the idea of helping people lower such hurdle, in this project, we tackled the problem of providing a pipeline that automatically display the dish images beside the queried dish names. With this application, people just need to simply put a menu in front of the camera, take a photo of the menu, let the program process the image and recognize the characters in the menu and finally present the images of the dishes of the original dish names on the menu in order to help the users have more straightforward understanding of the dishes they are interested.

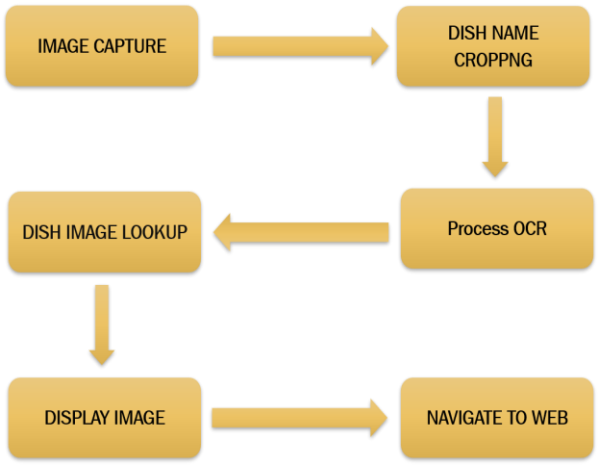
(II) RELATED WORK:

There are many implementations related to this topic, which gave us a great insight and inspiration when forming our idea. A. Heng described an iPhone application which is designed to quickly and easily split a restaurant bill amongst a group of people in his paper [1]. The application uses the Tesseract OCR engine to extract words from the receipt, then performs text processing to define individual items on the receipt. This application can effectively reduce the time of figuring out how much one has to pay from a group check. Based on the observation that paper receipt is still irreplaceable and there is no easy mean to convert it into an electronic format despite many advanced electronic payment systems exist, C. N. Nshuti discussed in his paper how to realize the digitalization of paper receipts and developed a pipeline for performing OCR on the picture of a document taken by a mobile phone [2].Our study focuses on developing different methods to increase the accuracy of the OCR algorithm, and then building a robust and real-time English menu translating system for non-English speakers.

We implemented several techniques we learned in class. The technical approach pipeline is described in section (III). We are concerned about the effect of different preprocessing techniques. As such, we performed detailed comparison of the results, which is described in section (IV).

(III) TECHNICAL APPROACH:

The application pipeline contains 6 main procedures: inputting the menu image (photo of the menu), Cropping Dish name then process it using Tesseract OCR then Dish image look up from the Database and finally display the image.



A: Image Capture

The input phase is quite straightforward: a user takes a photo of the menu with a camera or take an image from gallery. Photos taken in the natural scene have a great diversity and significant uncertainty, which leads to difficulty of recognizing characters from them. For example, characters in different images may have different sizes, fonts, colors, rotations and so on. There are other variables that may result from the environment such as motion blur and out out-of-focus image. These are definitely challenging problems, but we assumed that the user would have full control over the image quality at the time of image capturing; hence not part of our focus of the project.

B: Dish Name cropping

To facilitate and improve the accuracy from subsequent Optical character recognition (OCR) engine, we preprocessed the image by cropping for dish name and pass it into the OCR engine. There are two reasons to do it, First Tesseract OCR can't handle big images so cropping helps to OCR getting accurate results and second it helps to find one particular dish name.

C: Optical character recognition (OCR)

After getting dish name texts with in one single cropping box per line, we adopt implementation of Tesseract algorithm library, an open source OCR engine initially developed at HP Labs and currently managed by Google [4], in our project to perform character recognition. The basic principles of Tesseract OCR are as follows: First, character outlines are extracted and assembled together into Blobs by performing connected component analysis. Then the lines of text, which are formed by blobs, are split up into separate words depending on the spacing between each character. The following stage is known as word recognition, which is a two pass process. In the first pass each word is recognized in turn. Once

a word is recognized, it will be stored in an adaptive classifier and be used as training data. In the second pass the words that weren't successfully recognized are recognized again by using the training data obtained from the first pass. Finally, a string of words is given as an output.

D: Dish name matching with database

Dish name matching is performed after OCR string results are obtained. At this stage we may expect some misspellings coming from the OCR results, and we must correct them in order to find a match in our name list of image database. Here we utilize AutoComplete Text View, which increases the overall success rate of image lookup. We use the SQLite Database use for storing the dish names and images paths. SQLite is simple and internal database.

E: Display the final result

After looking up all recognized dish images from the database, finally we show the image of dish along with cropped image and Text. If is there any mistake in spelling you can change it using Text Field, it will give you Hint using AutoComplete Text View.

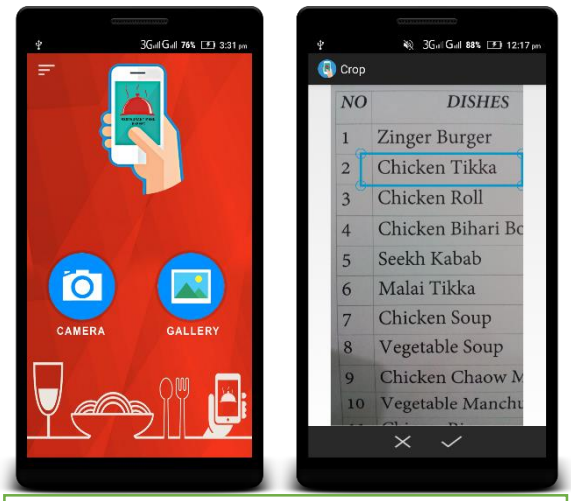
F: Navigate to web

After displaying final dish name user can navigate to the web by pressing on image. It will help to get more info about that dish user want to get. Text will go to the google search navigation and user can open it any browser.

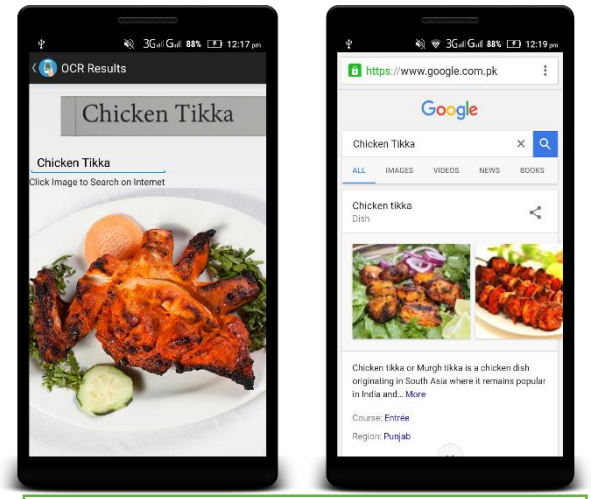
(IV). RESULTS

In this section, we present our experimental results of our pipeline and discuss the strengths and possible improvements to improve the performance.

A. Results of Entire Pipeline



A **B**



C **D**

Above figure shows the results of each step in our pipeline,
Incrementally adjusting the inputs into the OCR engine and
Fixing the types from the OCR result to displaying the final
Images of the dish names.

(V). FURTHER DISCUSSIONS AND FUTURE WORK

In this project, we have successfully developed an automatic English menu translation system to help non-English speakers overcome the difficulties of ordering meals in a foreign restaurant, with emphasis on studying different methods to increase the accuracy of the state-of-the-art OCR algorithm and finally increase the rate of correct recognition.
The system is robust, fast, accurate and flexible while providing good user interaction experience. With good scalability, the system can be further extended to and find more applications in situations such as multi-language translation and larger database, or the information about the dish can be obtained via online searching. People may further extend the application to showing the information on a wearable VR device.

REFERENCES

[1] A. Heng, “L’Addition: Splitting the Check, Made Easy”.
[2] C. N. Nshuti, “Mobile Scanner and OCR (A first step towards receipt to Spreadsheet)”.
[3] M. Jin, L. X. Wang, B. Y. Zhang, “Real Time Word to Picture Translation for Chinese Restaurant Menus,” EE268 Project Report, Spring 2014.
[4] R. Smith, “An Overview of the Tesseract OCR Engine,” Tesseract OSCON, Google Inc.