A Smart App for Mobile Phones to Top-Up User Accounts for Any Network Service Provider in SriLanka

C. L. Ishani S. Fonseka

Assistant Lecturer, Department of Mathematics, Eastern University Sri Lanka,

Abstract-

The objective of the project is to develop an Android application which supports mobile users to activate different services provided by service providers using activation codes in a user friendly and efficient manner. In this project, referred a number of USSD codes used by all the service providers which are available in Sri Lanka. Finally choose most common USSD codes for every service providers and generated this Application. Special feature of this Application is, take a photo of the re-charge card pin number and send it rather than manually typing that. For that purpose OCR was developed by using Android Studio to detect mobile recharge card pin number portion by doing image processing. It works with Tesseract Engine and tess-two library for OCR operation. After detection, it extracts the OCR text which is the pin number of recharge card and then it sends request to the respective mobile operator for recharging balance. This application have built an android based Mobile phone application so that the report can be worthy of real life experience.

Keywords- *Android, USSD codes , Tesseract Engine, Optical Character Recognition.*

I. INTRODUCTION

Smart phones are more common than computers today. Almost everyone in the world makes regular use of smart phones in their day to day lives. People can get a lot of different benefits from smart phones and that too in a very portable and mobile manner. All smart phones require an operating system as an interface and the most popular Operating System today is the Android. The android Operating System(OS) presents never before seen flexibility and support for third party applications. This has given rise to a huge amount of popularity of the Android Operating System not only among consumers, but also among developers. Android application development is fast becoming a separate field of Information Technology. More and more independent application developers and application development companies are taking interest in this Operating System and are coming up with some of the best applications.

Mobile applications designed to work with carrier provided services, and not utilize features of specific mobiles have generally focused on voice through Interactive Voice Response (IVR) and text through Short Message Service (SMS). IVR and SMS are Universal Apps in that they make services uniformly available on every mobile handset. There is a third option for a Universal App, which is less frequently used in Mobiles for Development, Unstructured Supplementary Service Data (USSD). USSD is a protocol defined inside the Global System for Mobile Communications (GSM) standard. USSD is a session based protocol, like IVR, that supports the exchange of text data, like SMS, thus filling a gap in the Mobile for Development design space. The most common uses of USSD are for customers interacting with a carrier's services, such as querying airtime, checking data balance or subscribing to information services. This is done by sending a star code (such as *100# to check credit balance) and accessing interactive menu systems. USSD has a number of advantages over SMS, such as providing greater privacy, which makes it a candidate for some important Mobile for Development applications. However, until recently, it has been very difficult to deploy USSD applications due to requirements of working directly with carriers.

At the moment the activation codes which are used to activate services need to be entered manually by the user. User has to remember all the activation codes and sometimes mobile users do not even know about certain activation codes provided by their service provider. If the user is using a dual sim mobile phone it is much more difficult for the user to remember the activation codes provided by different service providers. If the service provider has introduced new activation codes for new services user may not know about those new codes. And also for some activation codes, the user is required to enter a contact number and the code

manually. It would be more convenient for the user if the contact number can be searched from the list of contacts in the phone and add it in the activation code.

Mobile recharge cards are very essential for mobile phone users because of the increasing number of mobile users. Recharge card is much reliable than other techniques used to top up user accounts. It is not easy to keying the right pin number while anyone is walking or busy in work (numbers misplaced, less number typing). Most of them face difficulties keying the pin number correctly in one attempt. Many of us have ***presbyopia*** and the older people are often the victims of it. So, as a solution of that problem we propose to use an OCR to read pin numbers on scratch cards. This is a much better and efficient way to smartly recharge a mobile phone user accounts.

II. BACKGROUND

A. Android Operating System

The Android platform is a software stack for mobile devices including an operating system, middleware and key applications. Developers have full access to the same framework APIs used by the core applications.

B. Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools. Working with eclipse can be difficult at times, probably when debugging and designing layouts sometimes Eclipse get stuck and have to restart eclipse from time to time.

Android Software Development Kit (SDK) which includes a variety of custom tools that helps to develop mobile applications on the Android platform. The most important of these are the Android Emulator and the Android Development Tools (ADT) plug-in for Android Studio [1].

C. Optical Character Recognition (OCR)

Optical character recognition, usually summarized to OCR, is the mechanical or electronic translation of scanned images of handwritten, typewritten, or printed text into machine-encoded text. In order to extract and repurpose data from scanned documents, camera images, you need OCR software that would identify letters on the image and converted into words .

The Mobile OCR Engine is a software development kit (SDK) that allows developers to integrate optical character recognition technologies into Android apps that enabling them to convert images and photographs into editable and searchable text. It turns your mobile phone to text scanner. You can either implement your own OCR, or use an existing OCR tool / api.

There is an open source OCR library that supports android called Tesseract. It is good for, users who have trouble in sight and takes time to read the small printed material and users who have trouble in getting the numbers/letters.

Features of Image Scanner (OCR):

- Works offline .
- Smaller size.
- Extract Text/Numbers from Image.
- Edit and saved extracted text.
- Copy extracted text to Clipboard which can be used in any other applications.
- Save extracted text as text files.

D. Unstructured Supplementary Service Data (USSD)

USSD is a protocol used by GSM cellular telephones to communicate with the Service provider's servers.

III. APPLICATION OVERVIEW

This product needs access to the users contact list in the phone. Therefore it is somewhat dependent on the user's environment. And the android application is integrated with a web service which provides trainee data to the OCR when the camera is in it's first run. When the corresponding button is clicked, it is directed to its database and checks the slot 1 service providers name with the button name. Then it will give the requesting USSD code and finally it connects with the network operator. The output is a popup menu including requesting information and certain guides.

III. METHADOLOGY

. The main goal of this project is to develop a functional app that supports mobile users to activate services provided by mobile service providers in user friendly manner. Initially the system design has been divided into three basic components:

a. Design Graphical User Interface.
b. Design USSD Dialer
c. Design Optical Character Recognition.

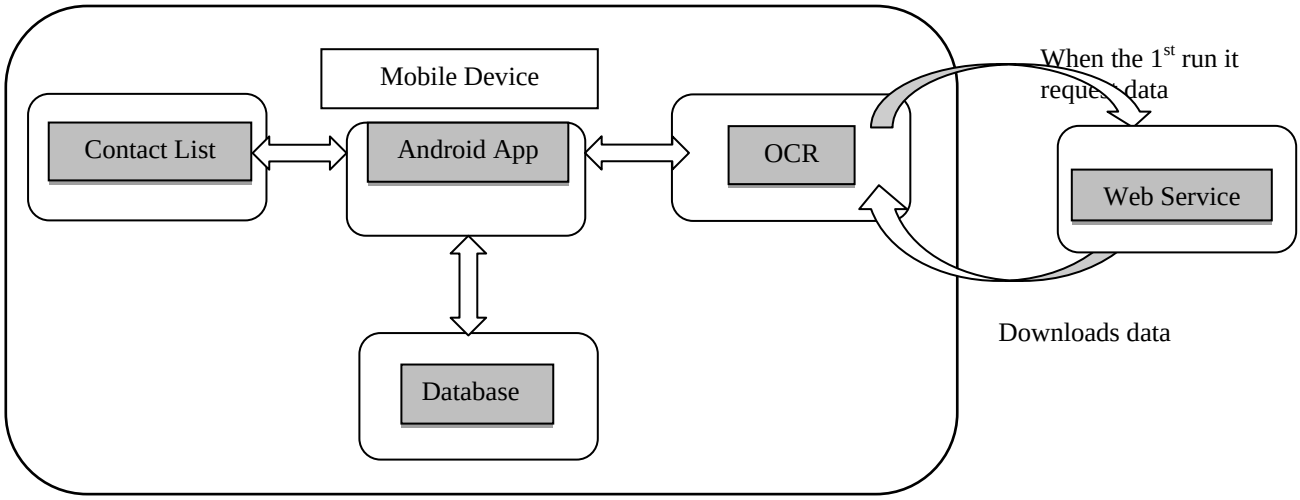


Fig 1: Application Overview

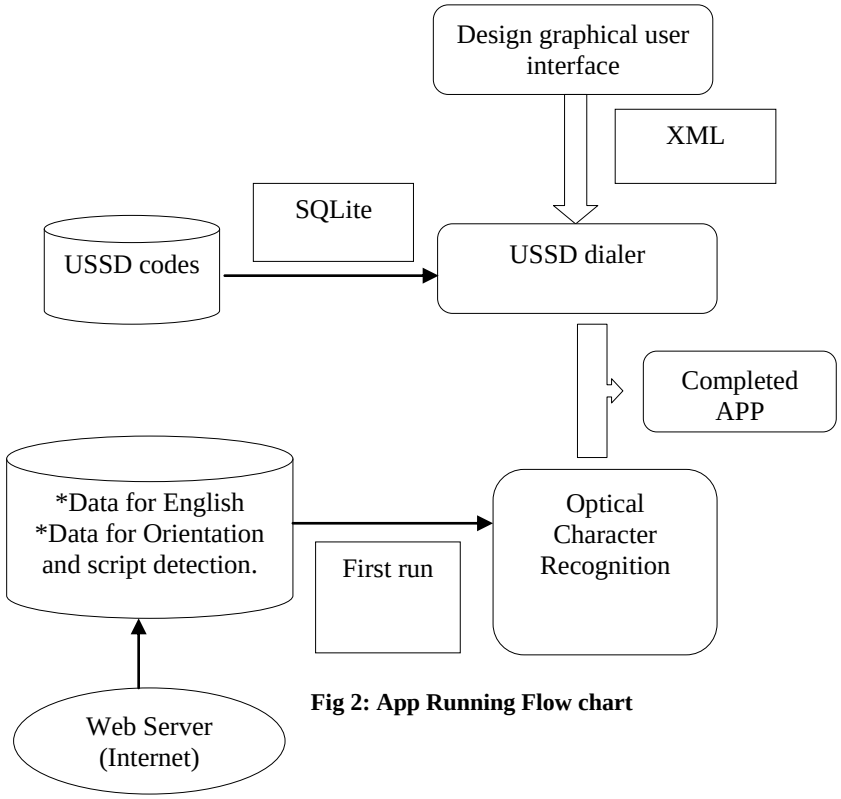


Fig 2: App Running Flow chart

A. *Designing graphical user interface*

As the very first step in designing, is design the user interface. An app's user interface is everything that the user can see and interact with. Android Studio offers an advanced layout editor that allows to drag-and-drop widgets into corresponding layout and previews that layout while editing the XML([1],[3],[10],[11]).

In this project basically we have the following user interface :

- Welcome page
- Home page
- Selecting page
- New Field page

} Mainly viewed as Tabs

- Details page
- Recharge
- Sharing

1) *Welcome Page and Home Page*

The welcome page is limited to be displayed only for 2 seconds (2000 milliseconds) and quickly it will direct the user to the Home page. It displays only the logo and the name of the application.

The home page detects the service provider of the available sim. (If the phone has Dual sim, it will detect slot 1 sim). Buttons are created dynamically with the help of database. It's design is very user friendly. It is simple and easy to understand all the operations without any guidance. Some buttons directly run corresponding to USSD code and special buttons are directed to a new page.

2) *Selecting Page and New Field Page*

Selecting page includes buttons for all the service providers available in Sri Lanka. When each button is selected it will direct to the details page. Each button is describe with corresponding service providers name and its official logo.

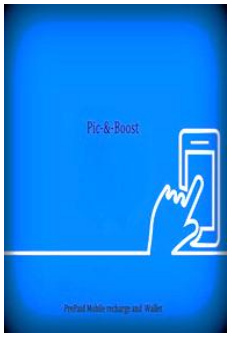
This page is provided for adding new button depending user's purpose. Then it will automatically add in to the database and appear in the home page. For deleting those newly added buttons user has to re-install the app again.

3) *Details Page , Recharge and Sharing*

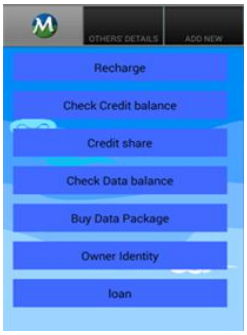
For each service provider it contains details of ussd codes which are commonly using in day today life. It is easy for user when he wants to deal with his/her slot 2 sim. Because this app only supports slot 1 sim.

For recharge purpose most of the peoples are using re-charge card by scratching the card user can get the pin number. That pin number can be manually entered or by capturing that pin number user can easily re-charge his/her account. And also it provides facility to send that pin number as a SMS to another phone. I t can recharge another account also by using that pin number. Camera button directs to the OCR. User also can access the contact menu directly.

If the user wants to share credits to another account Sharing page will help that purpose. It can directly access the contact menu and select the number instead of typing or remembering contact number. User no need to remember the USSD code for sharing. It automatically share credits when the user enter the correct information.



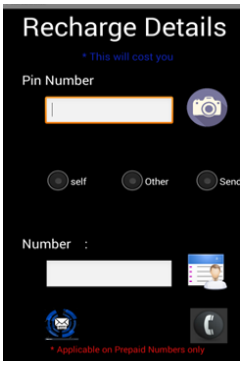
a. Welcome Screen



b. Main Screen

c. Service Provider Selecting

d. New Codes adding Screen



e. Credit Sharing Screen

f. Recharging Screen

Fig 3: User Interfaces

B. Design USSD Dealers

Collected USSD codes available in Sri Lanka and categorized them into two parts. Common USSD codes for all users and use only certain peoples when they want. Mostly using codes are again categorized as following figure. Select common codes as data for our project.

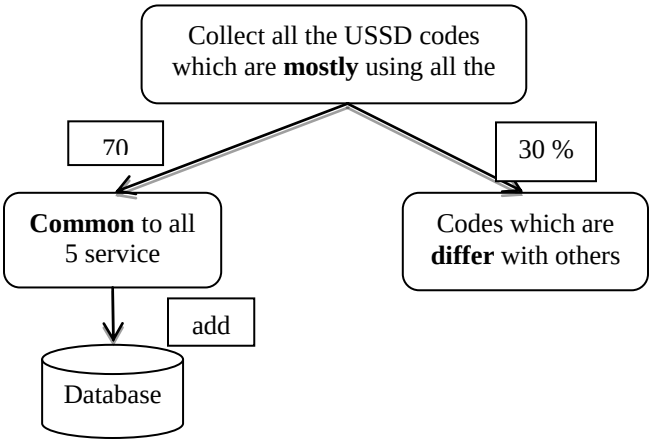


Fig 4: USSD Code Selection Chart

As an example, in Sri Lanka the following USSD code format is used in Mobitel(one of the Service Providers in Sri Lanka) to recharge.(*Ex:* Dial *448*03016650860471#)
" * " + " Operator-prefix " + " * " + "Pin number" + " # "

But using this top-up application it can be activate simply browsing through the menu rather than typing a long sequences of numbers. Then it send request to the service provider directly.

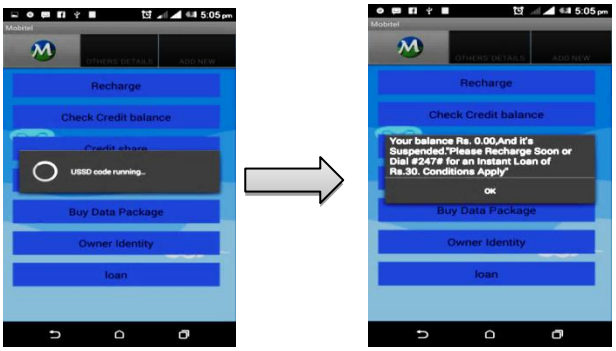


Fig 6: Notifications After Running USSD Code

C. Design Optical Character Recognition

Tesseract is an Open Source OCR engine, available under the Apache 2.0 license. It can be used directly, or (for programmers) using an API. It supports a wide variety of languages.

This project works with Tesseract v3.03. The required source code for Tesseract 3.03 and Leptonica 1.70
In this project we used tess-two which compiles the Tesseract and Leptonica libraries for use on the Android platform. It contains an Android library project that provides a Java API for accessing natively-compiled Tesseract and Leptonica APIs.

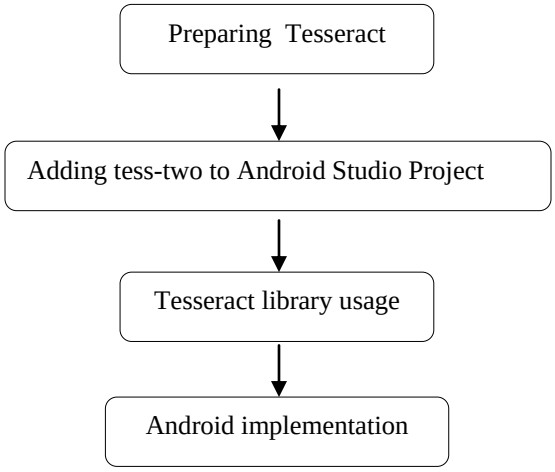


Fig 5: Activity Flow for the Implementation

In the very first run of this Application it downloads the data for English and then data for Orientation & script detection.

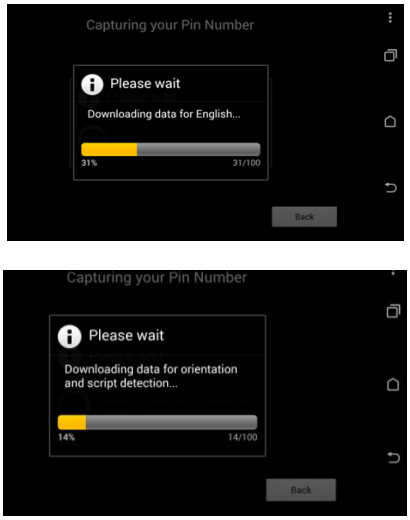


Fig 7: Downloading trained data

In TESSERACT OCR Engine module ([5]), The Captured Image takes place, after that the text layout is analyzed, detected and finally numbers and lines are detected. The words are sent to a number of passes. In these passes each word is chopped into characters and

characters are checked for the need of joining the broken characters or the breaking of associated characters. Finally sliced characters are identified with the help of inbuilt fuzzy features matched to language specific training data of Unicode characters. After each pass the words are matched back and forth with the Language specific Dictionary words.

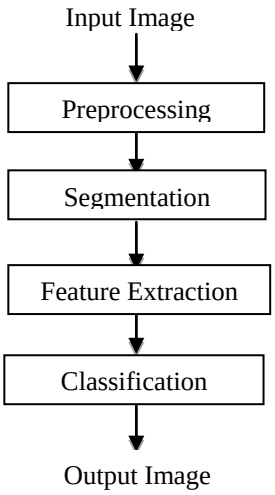


Fig 8: Optical Character Recognition Flow chart

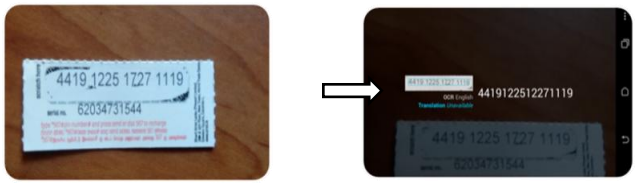


Fig 9: OCR Reads Pin Number

IV. EXPERIMENTAL RESULTS

This considers different criteria for and collecting and calculating result for different USSD Applications in order to compare the efficiency of the application under different conditions. To test that the OCR application is working correctly we used two criteria used. They are different light conditions in same device for different operators and in different devices for different operators in normal light condition. For OCR testing purpose 10 recharge cards were used for each service provider(10 * 5).

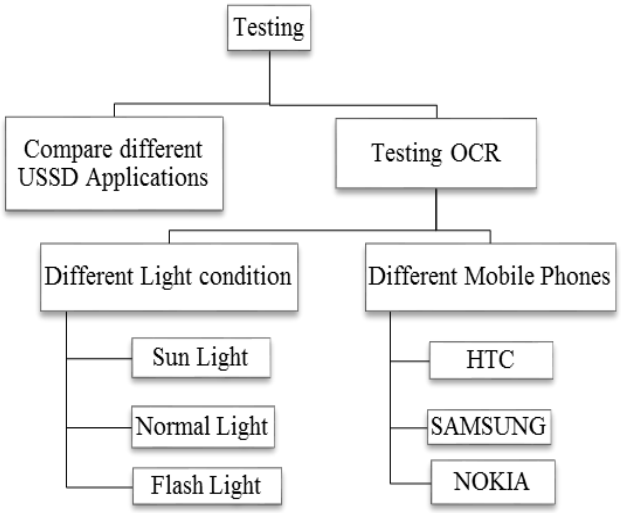


Fig 10: Application Tested on Different Conditions

A. Compare different USSD Applications

Collected common features among available different USSD Applications and compared those features with each and every applications including our Product (Pic & Boost).

$$\text{Qulity Precentage} = \frac{\text{No. of avaulable Features}}{\text{No. of considered Features}}$$

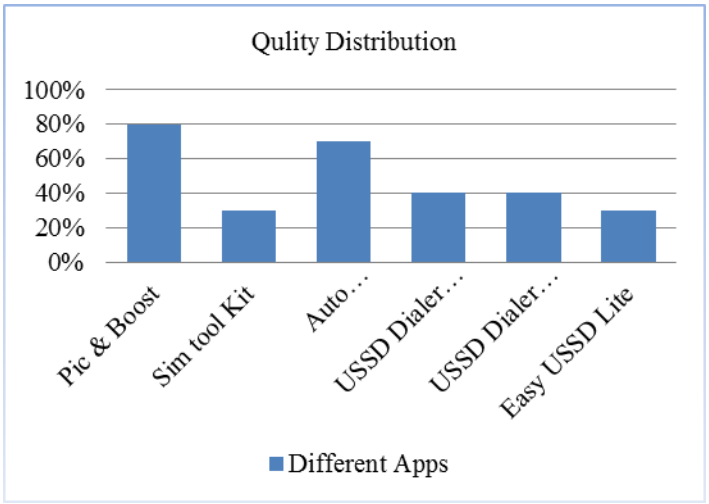


Fig 11: Quality Distribution between few USSD Applications

B. Different Light conditions

Initially we used 10 pre-paid recharge cards for each operator for testing the applications accuracy rate. This is done in the accuracy test in different light conditions : Sun light, Flash light and Normal light. Then we calculated the accuracy rate in terms of the

recognized pin numbers by tesseract from the total pin numbers that a recharge card have.

$$\text{Accuracy Precentage} = \frac{\sum \text{Correct pin numbers per card}}{\text{No. of pin numbers} * \text{No. of cards}}$$

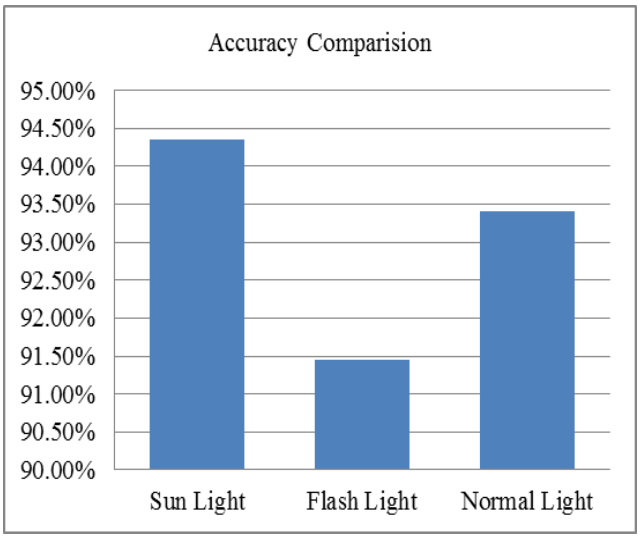


Fig 12: Accuracy Average in Different Light Conditions

V. CONCLUSION

In this project, we have investigated the performance of an app developed by as entitled "A Smart App for Mobile Phones to top-up user Accounts for Any Service Provider in Sri Lanka ". In this work we have proposed, designed, developed, implemented and tested the performance of this application. We have conducted tests under different testing conditions and have compared the performance of this application. The main feature of our app which is not available in other applications is the Optical Character Recognition part. This app can simply recognize re-charge card pin number by capturing the card number instead of manually typing it. We have carried out a number of testing cases and have collected the test results to compare the performance of our app.

To test the performance of the OCR app we have used 50 recharge cards and tested. And the other USSD code testing is countless. Because they are 100% correctly working. However it has some future development. Because it only works slot 1 sim in a Dual sim mobile phone.

REFERENCES

[1] Apply Android Studio (SDK) Tools by Hana R. Esmaeel ,Department of Inform. & Comm.Engg., Al-Nahrain University, Iraq(May 2015).

[2] USSD Services for Interactive Mobile Users Building User-Friendly Mobile Telephony

[3] Applications Using Dialogic® Distributed Signaling Interface Components (www.dialogic.com)

[4] USSD: The Third Universal App by Trevor Perrier , Brian DeRenzi

[5] Optical Character Recognition on Handheld Devices by Sravan Ch and ShivankuMahna .

[6] International Journal of Computer Applications (0975 – 8887) Volume 115 – No. 22, April 2015

[7] Android Travelmate Applications with OCR & Its Language Translation by Ashish Titus, Sagar Karkera, Suyog Oswal, Darshan Ingle ,SIESGST, Mumbai, University M.S., India March 2015

[8] Text Recognition and Translation Application for Smartphone, A. A. Tayade , Prof.R.V.Mante ,Dr. P. N. Chatur November 2013

[9] http://www.codeproject.com/Articles/476142/Optical-Character-Recognition

[10] http://www.intechopen.com/books/advances-in-character-recognition/online-handwrittenchinese-japanese-character-recognition

[11] Rice S., Nagy G., Nartker T., Optical CharacterRecognition: An Illustrated Guide to the Frontier,Springer, 1999.

[12] Code Gems For Android Developers Working with the World's Most Popular Mobile OS By Avi Yehuda (DZone Magazine)Authors