Interpretation of Tamil Script using Optical Character Recognition in Android

Arun Srivatsa.R , Shriya.P , Adithya Srinivasan

Final Year BE, Department of Computer Science and Engineering,
Meenakshi Sundararajan Engineering College,Chennai

***Abstract*— Tamil is one of the oldest surviving languages. Tamil poetry and prose encompass many intricate, thought provoking ideas. But the Tamil script is not widely known and is difficult to master. So, in order to make the Tamil language more accessible, we designed this application. This way Tamil text is easily available to those who are not familiar with it. The application captures the image of a Tamil script or handwritten Tamil script and digitizes it into Tamil font. Then translates it into English. This application is also designed to aid tourists to better experience Tamil Nadu by providing a quick point and shoot way of translating Tamil boards and other text.**

***Keywords*-- Image processing, Translation, Optical Character Recognition, Handwriting Recognition.**

I. INTRODUCTION

Tamil is one of the longest surviving classical languages in the world. Tamil has 12 vowels and 18 consonants. There are many Tamil literatures which are yet to be widely recognized by everyone. Tamil Nadu is one of the most popular tourist spots in India. Hence, we wanted to make Tamil language accessible to a much wider audience. Smart devices are owned by most people these days and what better way to promote a language than to have it at the tip of your fingers.

The first focus is to convert the text written in the real world setting into one which can be digitally manipulated. This is done using Optical Character Recognition. A document page is anything which contains characters printed by machine or is handwritten print or even cursive. The document page is the input for this application. Then OCR identifies the characters on a document page. [1,2].Tesseract Engine is used to digitize the content of the document page. After this has been done the identified text is fed to a translator and can be translated into English or any other preferable language.

II. OBJECTIVE

The main objective of this application is to make the Script of Tamil language more accessible. This application can be used in a variety of ways depending on the requirement of the user.

Tourists may not be able to understand what they see. But this application does. And this will help them to a greater extent and will allow them to be independent. In combination with other tourism apps, like maps, hotels etc., and this app will help in redefining tourism in Tamil Nadu.

Given a printed format of any Tamil work, any person with the basic knowledge of any one language can understand the content, by translating it to the required language.

This application can also be useful for archiving purposes. It is helpful in digitizing the articles either handwritten or printed by converting and storing it in a digital format.

III. MODULES

The application consists of three modules:

- Image Capture/ Load Image
- Processing using OCR
- Translation into English

Image Capture/Load Image:

The document page which can be Tamil text printed on a sheet of paper, sign boards, handwritten text or any non-digital format is captured using the inbuilt camera from the smart devices. For greater accuracy a camera with high resolution needs to be used. [7]

The document page can also be obtained by selecting the image from the storage area in the device. This is done by using the Load Image option. It accesses the internal or external memory of the smart device and this image is fed into the Tesseract Engine for processing. For example, a scanned copy of a Tamil Transcript.

And the document page which is obtained by either one of the above methods is now digitized using OCR technique.

Optical character recognition

Optical Character Recognition is used to detect the script from the image and converts it to digital font by using various algorithms like segmentation and, edge detection. Tesseract is the OCR engine that is used in this application. The image is first pre-processed. This includes proper alignment and smoothening to make the image recognition more accurate. Other pre-processing such as aspect ratio normalization and scaling are also done. [1,2]

After the pre-processing the Tesseract engine performs a two pass pattern matching. This involves comparing the image with stored characters pixel by pixel. However such a matching covers only a limited number of fonts. [3]

The accuracy can be greatly improved if we limit the number of characters. The characters which closely resemble each other can be eliminated by choosing the frequently used one and rejecting the least recently used character. In Tamil, for example, consider these two letters

கொ ள

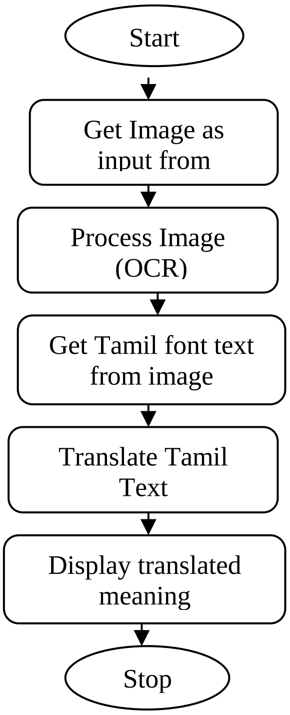
Though these two are completely different letters, the first characters are almost identical. Similarly, there are a considerable amount of characters that are very similar looking and this makes it difficult for the OCR engine to accurately identify the correct equivalent digital font for the given input image. So the accuracy in this case is compromised. However, the accuracy can be improved by training data for the input language. Here we have used an existing trained data for Tamil Language. [4]

Translation

The translation process is via a simple local database for the initial testing. The database consists of a limited number of words. A local database is used because many words in Tamil have multiple meanings according to the context in which they are used in. Since accuracy in translation cannot be achieved easily, we have a local database with limited words which are more frequently used. Theru (Street), Unavagam (Hotel) are some of the frequently used basic words. However, translation APIs can be used to improve the scope of words used in the application. Translation APIs will provide large collection of words which will help in making the translation part of the application more dependable.

IV. DATA FLOW

The following DFD represents the overall working of the application.



The camera is operated by clicking on the “capture” button found at the bottom of the front screen of the application. If a high resolution image already exists, then it can be retrieved from the Image gallery by choosing the image from the local internal or external storage. Once the image is captured or loaded it is input into the Tesseract Engine for processing.

The OCR first corrects the image in a number of ways to make it suitable for processing. The image is properly oriented if it is tilted. Then the lines and dots that are unnecessary are removed. The image is then converted to a binary coloured one i.e. black and white. After that the characters are isolated via segmentation. The segmentation is performed by aligning it to a uniform grid. [6]

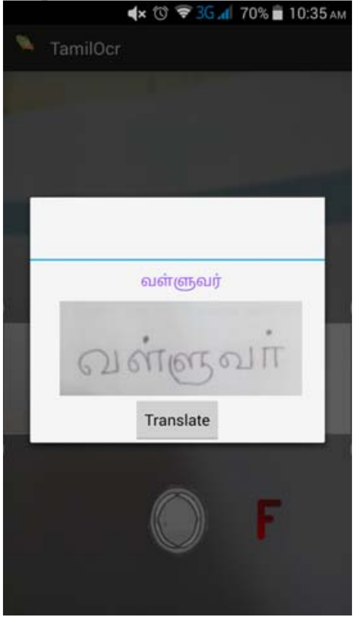
The OCR then processes the image with segmented characters by comparing it pixel by pixel with the stored characters. This process is known as pattern matching. [3] This is done by matching the segmented characters with the font in the trained data file. The Tesseract, though an excellent OCR engine, is not that adept at recognising the Tamil script initially. However by using certain training mechanisms this performance can be considerably improved.

V. RESULTS

The following figures show the results of the application.

Fig.1 Splash Screen

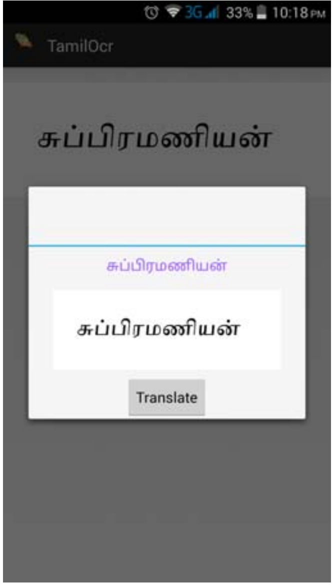
Fig 1 shows the splash screen of the Tamil OCR application.



வள்ளுவர்

Fig.2 OCR processing for Image from Camera

Fig 2 The image taken using the inbuilt camera of the smart device. And that image is fed into the OCR and its corresponding output is shown.



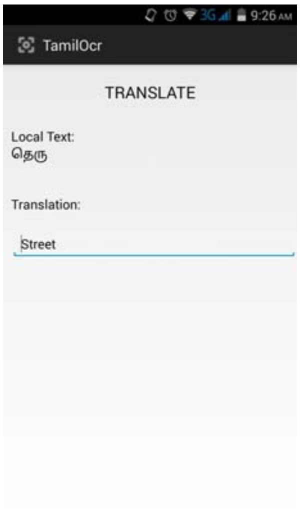
சுப்பிரமணியன்

சுப்பிரமணியன்

சுப்பிரமணியன்

Fig.3 OCR processing for Image from Local Storage

Fig 3 The image taken from the external storage of the smart device. And that image is fed into the OCR and its corresponding output is shown..



Local Text:
தெரு

Translation:
Street

Fig. 4 Translated Text

Fig.4 displays the translated text after matching it with the words in the local database.

VI. Conclusion

This application aims to make Tamil language more universally available. It has an easy interface to click pictures and get the translated text. While many languages have similar application, Tamil is yet to have one and hence this prototype is a step towards making Tamil more accessible. The main shortcomings of this application are: (i) The OCR processing of the Tamil Script is not accurate (ii) The translation can vary depending on the context of the language. Tamil is a complex language and hence training the Tesseract engine is complicated. However with proper and extensive training sets as well as a translation API, this project can be made more efficient and will be a great tool for interpreting and understanding the Tamil script.

References

[1] S. V. Rice, G. Nagy, and T. A. Nartker, Optical Character Recognition: An Illustrated Guide to the Frontier, Kluwer Academic Publishers, Norwell, MA, 1999 .

[2] Schantz, Herbert F., *The history of OCR, optical character recognition*. Manchester Center, Vt, 1982

[3] R. Smith. “An overview of the Tesseract OCR Engine.” Proceedings of 9th Int. Conf. on Document Analysis and Recognition, IEEE, Curitiba, Brazil, Sep 2007, p.629-633.

[4] Bhagvati, C, “On developing high accuracy OCR systems for Telugu and other Indian scripts,” Proceeding of Language Engineering Conference, 2002

[5] The Tesseract OCR engine, https://github.com/rmtheis/android-ocr

[6] Bissacco, A. ; Cummins, M. ; Netzer, Y. ; Neven, H., “PhotoOCR: Reading Text in Uncontrolled Conditions,” International Conference on Computer Vision, IEEE, 2013

[7] Optical character recognition in Android, http://www.codeproject.com/Tips/840623/Android-Character-Recognition