RESEARCH ARTICLE OPEN ACCESS

Embedded Optical Character Recognition On Tamil Text Image Using Raspberry Pi

V. Ajantha Devi[1], Dr. S Santhosh Baboo[2],
Ph.D & Research Scholar[1], Associate Professor[2]
Department of Computer Science
D.G.Vaishnav College,
Chennai Tamilnadu, India

ABSTRACT
Optical Character recognition is used to digitize and reproduce texts that have been produced with non-computerized system. Digitizing texts also helps reduce storage space. Editing and Reprinting of Text document that were printed on paper are time consuming and labour intensive. Optical Character recognition is also useful for visually impaired people who cannot read Text document, but need to access the content of the Text documents. This paper is on Methodology of a camera based assistive device that can be used by people to read Tamil Text document. The framework is on implementing image capturing technique in an embedded system based on Raspberry Pi board

Keywords:- Embedded System; Optical Character Recognition; image capturing; Tamil Text; Raspberry Pi board.

I. INTRODUCTION

Optical Character Recognition [6] has been an active subject of research since a decade. The rapid growth of digital libraries worldwide poses new challenges for document image analysis research and development.

Digital libraries offer access to larger document collection, and at a faster speed. A number of OCR software [12] – [16] available in market claims 99% recognition accuracy, but in practice these accuracy rates are rarely achieved. Most systems breakdown when input document images are highly degraded.

OCR is a process which associates a symbolic meaning with objects (letters, symbols an number) with the image of a character. It is defined as the process o converting scanned images of machine printed into a computer process able format.

A methodology is implemented to recognition sequence of characters and the line of reading. As part of the software development [11] the Open CV (Open source Computer Vision) libraries is utilized to do image capture of Tamil text, to do the character recognition. Optical character recognition (OCR) is the translation of captured images of printed Tamil text into machine-encoded text. It is widely used to convert books and documents into electronic files for use in storage and document analysis. OCR makes it possible to apply techniques such as machine translation, text-to-speech and text mining to the capture / scanned page.

II. TAMIL LANGUAGE

Tamil is an Indian language spoken widely in TamilNadu in India. Like all other Indian languages the basic structure of

Tamil script [10] is different from Roman Script. In particular, Tamil OCR is more complicated than other related works. This is because Tamil letters as in Fig 1. have more angles and modifiers. Additionally, Tamil script contains large number of character sets.

Fig 1. Tamil Letters

A total of 247 characters; consisting of 216 compound characters, 18 consonants, 12 vowels and one special character. Challenges that researches face during recognition process are due to the curves in the characters, number of strokes and holes, sliding characters.

A. Tamil Unicode

The Unicode Standard (http://www.unicode.org) is the Universal Character encoding scheme for written characters and text. It defines the uniform way of encoding multilingual text that enables the exchange of text data internationally and creates the foundation of global software. The Tamil Unicode [4] range is U+0B80 to U+0BFF. The Unicode characters are comprised of 2 bytes in nature. For example, the Unicode for the character is 0B85; the Unicode for the character is

0BAE+0BC0. The Unicode is designed for various other Tamil characters

III. SYSTEM HARDWARE DESIGN

The Hardware system is composed by following parts: an image capturing camera [27] [35], Raspberry Pi board to run image recognition programs on it and a Headphone to deliver the output speech. The system block diagram is shown in Fig 2.

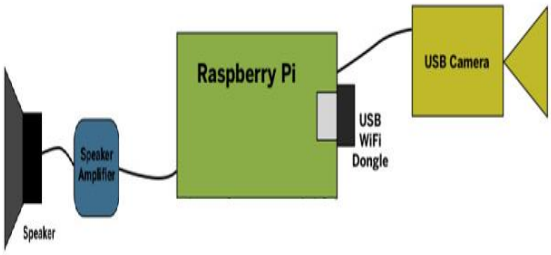


Fig 2. System Hardware Block Diagram

A. *Raspberry pi*

The **Raspberry Pi** [2] is a credit card sized single-board computer developed in the UK by the Raspberry Pi Foundation with the intention of stimulating the teaching of basic computer science in schools. Raspberry Pi is constructed for Broadcom's BCM2835 system around the circuit, which includes a 700 MHz ARM11 family processor and include 250 MHz clock -frequency Broadcom's Video Graphics Core IV. Memory B-model is 512 MB, and it is divided into the graphics card.

B. *Raspberry pi Camera*

The camera module used in this project is Raspberry PI NOIR (No IR) CAMERA BOARD [2] as shown in the Fig 3. The camera plugs directly into the Camera Serial Interface (CSI) connector on the Raspberry Pi. It's able to deliver clear 5MP resolution image, or 1080p HD video recording at 30fps.

Fig 3. Raspberry PI NOIR (No IR) CAMERA BOARD

The module attaches to Raspberry Pi, by way of a 15 pin Ribbon Cable, to the dedicated 15 pin MIPI CSI, which was designed especially for interfacing to cameras. The CSI bus is capable of extremely high data rates, and it exclusively carries pixel data to the BCM2835 processor.

A. *Enabling the camera*

Open the *raspi-config* tool from the Terminal:

sudo raspi-config

1. Select Enable camera and hit Enter,
2. Then go to Finish and you'll be prompted to reboot.

This camera board which has no infrared filter making it perfect for taking infrared photographs or photographing objects in low light (twilight) conditions. Other features of this camera board are Automatic image control functions Programmable controls for frame rate 32 bytes of embedded one time programmable (OTP) memory and Digital video port (DVP) parallel output interface Excellent

C. *Take a picture*

The following code will take a single picture and save it to 'foo.jpg'.

```
import time
import picamera
with picamera.PiCamera() as camera:
  camera.resolution = (1024, 768)
  camera.start_preview()
  # Camera warm-up time
  time.sleep(2)
  camera.capture('foo.jpg')
```

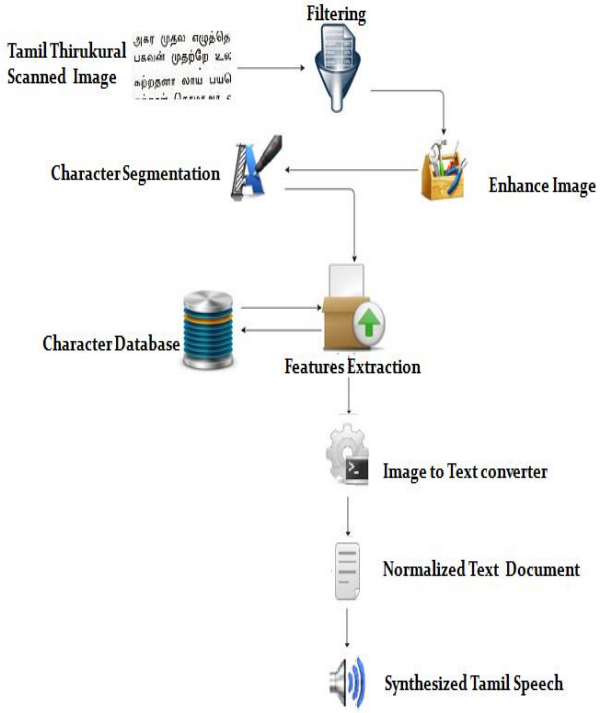
D. *Storage (Memory)*

The design does not include a built in hard disk or solid state drive, instead relying on an SD card for booting and long term storage. This board is intended to run Linux kernel based operating systems. This Raspberry Pi module has a Samsung class 4 micro SD card preloaded with the official Raspberry Pi NOOBS (New Out of Box Software) package, and a beautifully screen printed Micro SD card adaptor. The system designed can be operated in two sessions, i.e. one for capturing and creating a data base and the other session is to capture the image and which can be used for identifying or comparing the images in the database.

IV. SYSTEM SOFTWARE DESIGN

The Fig 4 shows the system to translate a Tamil Image into Tamil script and represents the converted script as normalized text and to voice output to the user passes through the various stages.

The various phases are scanning phase (image capturing), preprocessing, segmentation, feature extraction, Unicode mapping[4] and conversion of normalized text and finally to audio output, applying pattern matching algorithm to get equivalent text and voice file of matched Tamil Thirukkural stored in database.



.

Fig 4. System Software Block Diagram

The test image was captured by using Raspberry PI NOIR (No IR) CAMERA BOARD and the image store in JPEG format.

A. *Image Capturing*

The first step in which the device is moved over the printed page and the inbuilt camera captures the images of the text. The quality of the image captured will be high so as to have fast and clear recognition due to the high resolution camera.

B. *Pre-processing*

Preprocessing [3] [10] stage consists of three steps: Skew Correction, Binarization and Noise removal. The captured image is checked for skewing. There are possibilities of image getting skewed with either left or right orientation. Here the image is first brightened and binarized.

The function for skew detection checks for an angle of orientation between ±15 degrees and if detected then a simple image rotation is carried out till the lines match with the true horizontal axis, which produces a skew corrected image. The noise introduced during capturing or due to poor quality of the page has to be cleared before further processing.

C. *Segmentation*

After pre-processing, the noise free image is passed to the segmentation phase. It is an operation that seeks to decompose an image of sequence o characters into sub-image of individual symbol (characters). The binarized image is checked for inter line spaces. If inter line spaces are detected then the image is segmented into sets of paragraphs across the interline gap. The lines in the paragraphs are scanned for horizontal space intersection with respect to the background. Histogram of the image is used to detect the width of the horizontal lines. Then the lines are scanned vertically for vertical space intersection. Here histograms are used to detect the width of the words. Then the words are decomposed into characters using character width computation

D. *Feature Extraction*

Feature extraction [4] [8] is the individual image glyph is considered and extracted for features.
First a character glyph is defined by the following attributes: (1) Height of the character; (2) Width of the character; (3) Numbers of horizontal lines present—short and long; (4) Numbers of vertical lines present—short and long; (5) Numbers of circles present;(6) Numbers of horizontally oriented arcs; (7) Numbers of vertically oriented arcs; (8) Centroid of the image; (9) Position of the various features; (10) Pixels in the various regions.

E. *Image to Text Converter*

The ASCII values of the recognized Tamil characters are processed by Raspberry Pi board. Here each of the characters is matched with its corresponding template and saved as normalized text transcription. This transcription is further delivered to audio output.

F. *Text to Speech*

The scope of this module is initiated with the conclusion of the receding module of Character Recognition. The module performs the task of conversion of the transformed Tamil text to audible form.

The Raspberry Pi has an on-board audio jack, the on-board audio is generated by a PWM output and is minimally filtered. A USB audio card can greatly improve the sound quality and volume. Two options of attaching a microphone into Raspberry Pi. One is to have USB mic, another to have an external USB sound card.

To enable USB audio output, load the sound driver:

sudo modprobe snd_bcm2835

Enable USB audio output by default

sudo nano /etc/asound.conf

To playback: *aplay test.wav*

To adjust some volumes: *alsamixer*

V. EXPERIMENT ANALYSIS

The methodology used in this paper is analysised by using the Tamil image processing on Thirukkural Book.

Thirukkural is a classic Tamil sangam literature consisting of 1330 couplets or Kurals. It was authored by a Jain ascetic Thiruvalluvar, a poet who is said to have lived anytime between 2nd century BCE and 5th century CE. This methodology particularly apparent in the schooling system. Following are the snapshots on result of the analysis.

A couplet or *Kural* consists of seven *cirs*, with four *cirs* on the first line and three on the second. A *cir* is a single or a combination of more than one Tamil word. For example, *Thirukkural* is a *cir* formed by combining the two words *Thiru* and *Kural*, i.e. *Thiru + Kural = Thirukkural*

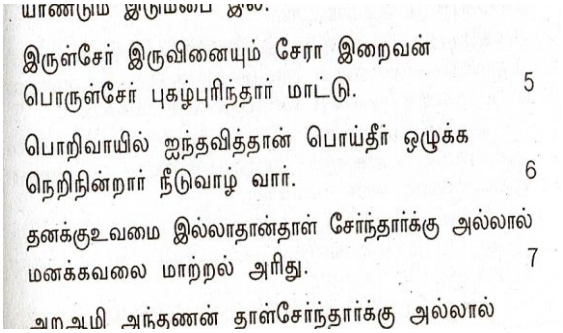


Fig 5. image of Tamil Kural

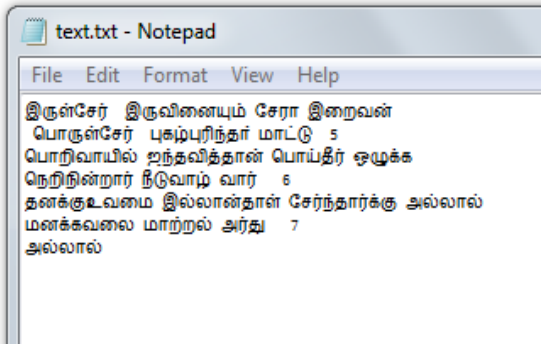


Fig 6. Text conversion of the Fig 5

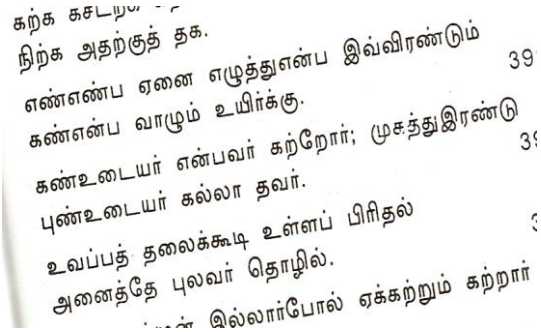


Fig 7. image of a tilted document

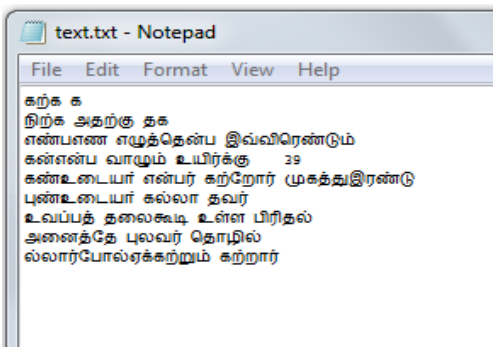


Fig 8. Text conversion of the Fig 7

The Final process of the above methodology is complete as the above text is synthesized and converted in to audio format and played using microphone or mini speaker connecting to on-board audio jack of Raspberry Pi.

VI. CONCLUSION

The device [11] with the methodology detailed in this paper helps the visually challenged to read books. The present implementation is only for the conversion of printed Tamil text. The device will be held like a image capture camera and captured over a printed page.

The input is taken by a camera in the hand-held [9] device and the output is given as speech through microphone using the above hardware interface. The software implementation and the hardware interfacing were done using Embedded Linux[1].

VII. ACKNOWLEDGEMENT

The author wish to thank the Christian Foundation for the Blind- India, Chennai for the support. The above paper is part of UGC Major Research Project of the Principal Investigator Capt. Dr. S Santhosh Baboo.

VIII. REFERENCE

[1] http://www.linuxcircle.com

[2] www.raspberrypi.org

[3] K.H.Aparna, V.S.Chakravarthy (2003), "A Complete OCR System Development of Tamil Magazine Documents", Tamil Internet 2003, Chennai, pp. 45 – 51.

[4] SEETHALAKSHMI R., SREERANJANI T.R., BALACHANDAR T., Abnikant Singh, Markandey Singh, Ritwaj Ratan, Sarvesh Kumar (2005), "Optical Character Recognition for printed Tamil text using Unicode", Journal of Zhejiang University SCIENCE ISSN 1009-3095, pp1297 – 1305.

[5] Akshay Apte and Harshad Gado, "Tamil character recognition using structural features" ,2010

[6] Kanimozhi.V.M, Muthumani.I (2014), "Optical Character Recognition for English and Tamil Script", International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 1008-1010, pp 1008 – 1010.

[7] G. Siromony, R. Chandrasekaran, M. Chandrasekaran, "Computer Recognition Of Printed Tamil Characters", Pattern Recognition (1978) 243-247.

[8] S.T.Deepa, Dr.S.P.Victor (2012), "Tamil Text Extraction ",International Journal of Engineering Science and Technology, Vol. 4 No.05 May 2012, pp 2176 – 2179.

[9] T.Rubesh Kumar, C.Purnima, " Assistive System For Product Label Detection With Voice Output For Blind Users ", IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 1, Issue 6, Dec-Jan, 2014, pp 1- 4.

[10] Aparna K G and A G Ramakrishnan , "A Complete Tamil Optical Character Recognition System",

[11] Jacob George, Varghese George, A. Deepan, Joshua Ninan Valson, "Optical Character Recognition Based Hand-Held Device for Printed Text to Braille Conversion", 2011 3rd International Conference on Machine Learning and Computing (ICMLC 2011). pp 172- 175.

[12] Dan ClaudiuCires¸an and Ueli Meier and Luca Maria Gambardella and JurgenSchmidhuber, "Convolutional Neural Network Committees for Handwritten Character Classification", 2011 International Conference on Document Analysis and Recognition, IEEE, 2011.

[13] GeorgiosVamvakas, Basilis Gatos, Stavros J. Perantonis, "Handwritten character recognition through two-stage foreground sub-sampling" ,Pattern Recognition, Volume 43, Issue 8, August 2010.

[14] Shrey Dutta, Naveen Sankaran, PramodSankar K., C.V. Jawahar, "Robust Recognition of Degraded Documents Using Character N-Grams", IEEE, 2012.

[15] Naveen Sankaran and C.V Jawahar, "Recognition of Printed Devanagari Text Using BLSTM Neural Network", IEEE, 2012.

[16] Yong-Qin Zhang, Yu Ding, Jin-Sheng Xiao, Jiaying Liu and Zongming Guo1, "Visibility enhancement using an image filtering approach", Zhang et al. EURASIP Journal on Advances in Signal Processing 2012.