IJCTA, 9(10), 2016, pp. 4809-4818
© International Science Press

Portable Device Converting Braille and Text Document to Speech Output Using Raspberry Pi

***Dr. V.Ajantha Devi.**

Abstract : Optical Braille Recognition and Optical Character Recognition is used to digitize Braille script and Text document that have been produced with non-computerized system. Digitizing Braille documents and Text documents also helps reduce storage space. Editing and Reprinting of Braille document and Text document that were embossed and printed on paper are time consuming and labour intensive. Optical Braille recognition and Optical Character recognition are useful for visually impaired people who cannot read Text document and Braille script, but need to access the content of the documents. This paper is on Methodology of a camera based assistive device that can be used by visually impaired people to get information from Tamil Braille document and Tamil Text document by Speech output. The framework is on implementing image capturing technique in an embedded system based on Raspberry Pi board. The post processing of the system is recognition of Tamil Braille document and Tamil Text document and converting the text to Audio format technique in an embedded system based on Raspberry Pi board.

Keyword : Embedded System; Optical Braille Recognition; Optical Character Recognition; image capturing; Tamil Text; Tamil Braille System; Raspberry Pi board.

1. INTRODUCTION

The Braille script invented in 1825 by Louis Braille is used by visually impaired people to read and write. The preferred means of communication for a visually impaired is through Braille system [11]. With the invention of computers, several approaches to computerized Braille translation [5] – [7] have been developed and thus this makes computerized Braille [9] is a recently new phenomenon with the Braille script. Many applications have been created for translation from Braille [10] to Speech. However, there is a shortage of programs designed for Tamil Braille. Optical character recognition (OCR) [14] is the translation of captured images of printed text in books and documents into machine-encoded text. OCR makes it possible to apply techniques such as machine translation, text-to-speech and text mining to the capture / scanned page. A number of OCR software [12] [18] available in market claims 99% recognition accuracy, but in practice these accuracy rates are rarely achieved for Tamil text [13] [14].

The written communications barrier between visually impaired persons and sighted is particularly apparent in the schooling system, nowadays, visually impaired students are taught in mainstream classes. Many of these students perform assessment, tests and homework writing using the Braille script. However, most staff of these students is Braille illiterate. Problems also exist in the workplace where any information embossed by a Braille user that is to be interpreted by other Braille illiterate persons, needs to be first translated by the Braille user. To overcome this

* Assistant Professor, Dept of CS & IT, Sri Adi Chunchanagiri Women's College, Cumbum –Theni Dt.,Tamil Nadu, India. ajay_press@rediffmail.com

communications barrier, the prototype developed provides a means by which Braille illiterate persons can read what a Braille user has written and vice versa[17]. The device is essentially a hand held scanner that capture the image of the Tamil Braille scripts or Tamil text documents and converts into equivalent decoded Tamil text which is capable of doing the extraction of Tamil Braille Character from a Braille document followed by decoding them into Tamil Characters and then the decoded characters are normalized to Tamil text and finally converted into audio output[23].

2. TAMIL LANGUAGE

Tamil is a language spoken widely in TamilNadu in India. The basic structure of Tamil script [17] is different from Roman Script. In particular, Tamil OCR is more complicated than other related works, because Tamil letters as in Figure 1 have more angles and modifiers. Additionally, Tamil script contains large number of character sets.

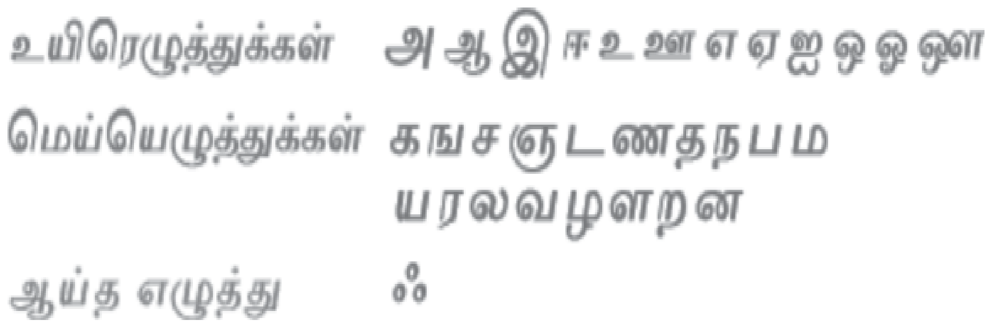

Fig. 1. Tamil Letters.

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 [12].

A. Tamil Unicode

The Unicode Standard (http://www.unicode.org) is the Universal Character encoding scheme for written characters and text. The Tamil Unicode [13] range is U + 0B80 to U + 0BFF. The Unicode characters are comprised of 2 bytes in nature. It defines the uniform way of encoding multilingual text that enables the exchange of text data internationally and creates the foundation of global software.

B. Tamil Braille System

The Braille script comprises of a "cell" per character and consists of six embossed or raised dots arranged in a rectangle containing two columns of three dots each [2] as shown in Figure 2 the positions being universally numbered 1 through 3 from top to bottom on the left, and 4 through 6 from top to bottom on the right. A dot may be raised at any of the six positions to form 2^6(64) combinations (including the combination in which no dots are raised).

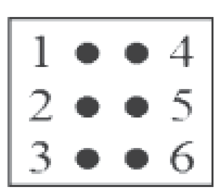

Fig. 2. Braille cell.

Because of its simplicity, comfortable for visually impaired to use it for read and write the Braille code system has become widely used by several communities. Braille script was applied or translated into several languages including Tamil language as shown in Figure 3.

Fig. 3. Tamil Braille cell.

The vowels and consonants combine to form 216 compound characters, as this combination turns out be in big number, two Braille cells are needed to represent these combination. Thus only 38 cells have been assigned for Tamil Braille[25][26].

3. SYSTEM HARDWARE DESIGN

The methodology is composed by following parts: an image capturing camera, Raspberry Pi board to run image recognition programs and a Headphone to receive the output speech. The system block diagram is shown in Figure 4.

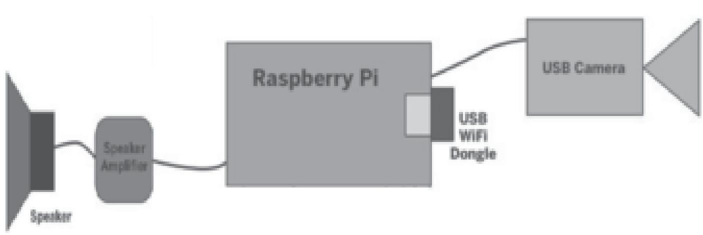

Fig. 4. System Hardware Block Diagram.

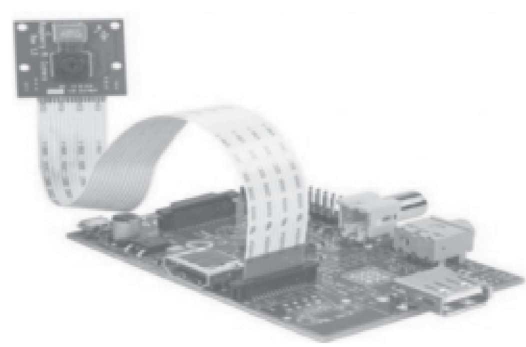

Fig. 5. Raspberry PI NOIR CAMERA BOARD

(*a*) Raspberry pi

A credit card sized single-board computer the **Raspberry Pi** [21] is developed in the UK by the Raspberry Pi Foundation with the intention of stimulating the teaching of basic computer science in schools.

Raspberry Pi contains a comprehensive amount of features, 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

Raspberry PI NOIR CAMERA BOARD is used in this prototype as shown in the Figure 5. The camera plugs directly into the Camera Serial Interface (CSI) connector on the Raspberry Pi able to deliver clear 5MP resolution image, or 1080p HD video recording at 30fps.

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.

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*) 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 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 used for identifying or comparing the images in the database.

4. SYSTEM SOFTWARE DESIGN

The system is designed to translate a Tamil Braille script and Tamil text document as Image into Tamil script and represents the converted script as normalized text and to voice output as in Figure 6.

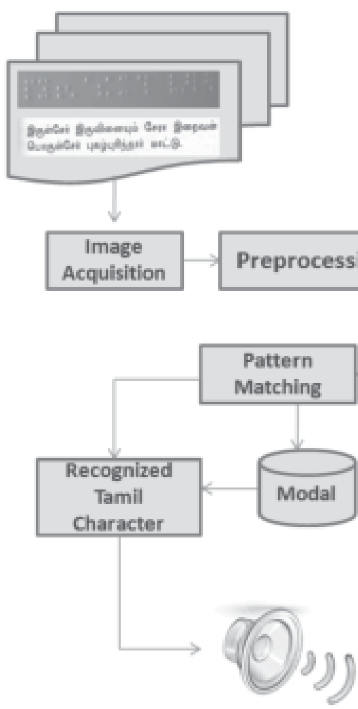

Fig. 6. System Software Block Diagram (Braille/Text to Speech).

The user passes through several stages for *Braille document image* including image capturing or scanning, converting the Image to Gray Level, image Thresholding (converting it to binary), de-skewing the image, cropping the cell frames, cropping the dot frames from cell frames, generating binary equivalent of activated Braille dots, generating equivalent decimal Braille code, apply matching algorithm to Braille frames, get equivalent text and voice file of matched Braille cell.

The various phases for *Tamil text document* 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.

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

1. Braille Image processing

Braille script document [3] [4] is a page that written using Braille alphabets. In this prototype the design based on standard specification of Tamil Braille script.

Fig. 7. Scanned test image.

Fig. 8. Gray level of Figure 7.

(*i*) **Converting the Image to Gray Level :** Inside a computer system, colored images are stored in 3-D arrays while gray level images are stored in 2-D arrays. Dealing with 2-D arrays is much easier and faster. Therefore, the first step in the proposed system converts colour scanned images to gray level so that any pixel value in the image falls within the range 0-255 as could be seen in Figures 7 and 8.

(*ii*) **Cropping the Image Frame :** Some of the scanned images suffer from having either black or white frames that would affect the thresholding step coming next. It is a must that those frames are cropped before proceeding forward. To do that, the average gray level is calculated for the whole image and then for each of the rows and columns separately.

(*iii*) **Image Thresholding :** To handle the thresholding Figure 9 an algorithm was developed to which works as follows; after converting the image to gray level and removing highest and lowest values to leave out distinct values, the average gray level for the whole image is calculated.

Fig. 9. Image after adaptive thresholding.

(*iv*) **Image De-Skewing :** Thus a binary search algorithm was developed to correct any de-skewing in tilted scanned images. The maximum degree of recognizing a de-skewed image as in Figure 10 is 4 degrees from either the left or the right side.

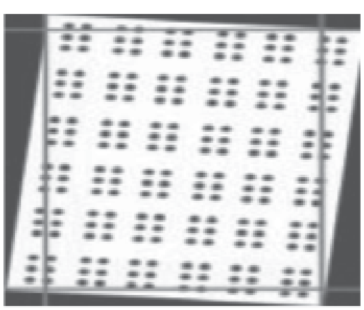

Fig. 10. Crop effect on skewed image

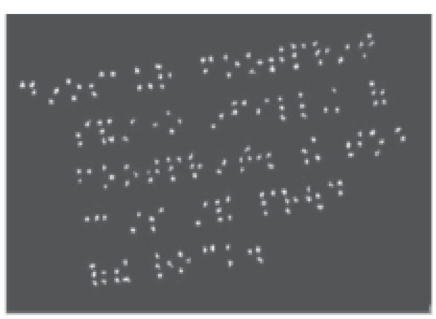

Fig. 11. Crop effect on skewed Braille image

Fig. 12. Image after cropped and skewed

After determining the deviation degree, rotate the original gray image and then Horizontal projection is performed to count the number of pixels in each row for the selected part in the previous step. Then the rows having more than 10 pixel points was calculated.

The image is rotated 4 degrees one time to the left and one to the right for the rotated image. This is done because if the image is skewed then cropping may remove some of the image parts that constitute Braille cells as could be seen in Figure 12.

(*v*) Dot Parts Detection : Experiences have shown that detection of Braille dot parts is better than detection of the whole dot. It was also proven that the average dot height is 8 pixels. Each dot is composed of a bright and a dark region with a small space between them as in Figure 13. The testing image analyzed by using Single side Braille Document.

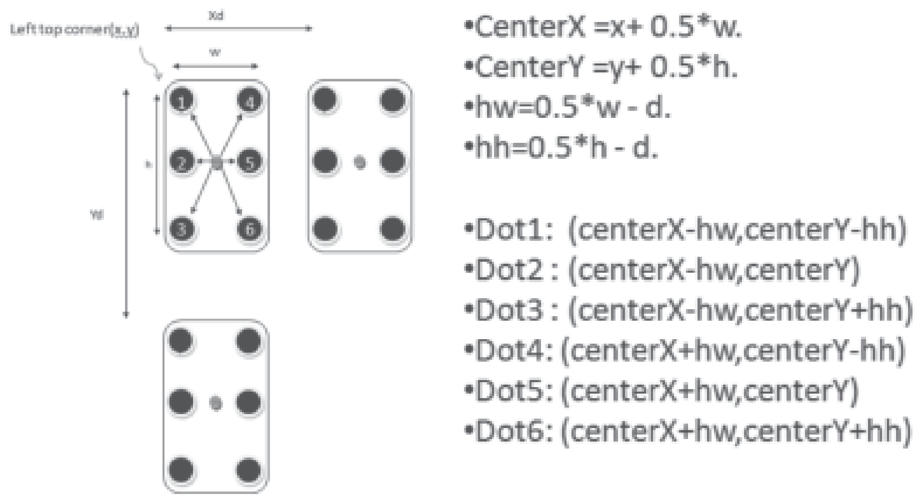

Fig. 13. Dot part detection technique

(*vi*) Whole Dot Detection : The framing process to determine the cells, words and lines in the scanned image based on standard dimensions of Braille documents as shown in Figure 14 , the cell frame size is known and experimentally recorded as 95X80 pixels in resolution of 200 dpi. Also the dot size recorded as 20X15 pixels in resolution of 200 dpi.

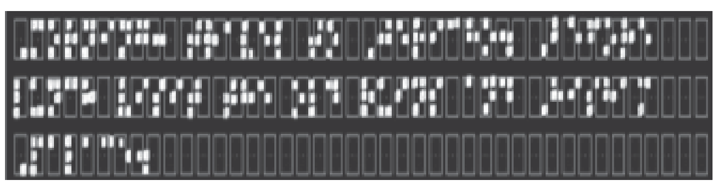

Fig. 14. Shows the dots at the end of the search.

(*vii*) Braille letter recognition and transcription : In this stage of the phase the Braille letter was recognized using matching algorithm to match each of the input decimal Braille code [8] from an input processed image with codes of each Tamil letter. Then the image is converted to its decimal code representation:

$$\text{Decimal-Code} = b1 + b2*2 + b3*4 + b4*8 + b5*16 + b6*32$$

After recognition process implemented the recognized letter transcript into equivalent text file through addressing process to run matched equivalent files from stored addressed database.

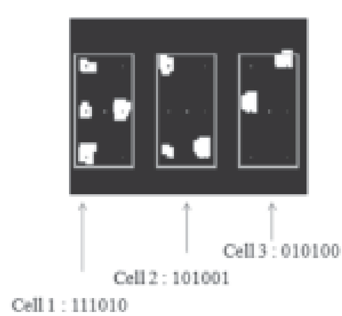

Fig. 15. ASCII Code for each cell.

(*viii*) **Conversion to text :** This is the last stage in implementation, the word recognition process flow through letter recognition and fill Braille decimal array of word, to apply matching process with stored addressed text files of words, then to run equivalent files from addressed word database.

Each Braille character is coded into a six bit binary number based on the presence or absence of dots. The presence of a dot is indicated by 1 and its absence by 0 as in Figure 15. This step is a simple one to one mapping from a database containing the binary code and the letter corresponding to this representation.

2. Text Image processing

(*i*) **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.

(*ii*) **Pre-processing :** Preprocessing [12]-[17] 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.

(*iii*) **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 Figure 16. 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.

Fig. 16. Line segmentation

(*iv*) **Feature Extraction :** Feature extraction [13] [15] is the individual image glyph is considered and extracted for features. First a character glyph is defined by the following attributes: Character Height; Character Width; Counts of horizontal lines present; Counts of vertical lines present; Total numbers of circles present; Counts of horizontally oriented arcs; Counts of vertically oriented arcs; Identifying the Centroid of the image; Position of the various features; Pixels in the various regions.

(*v*) **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.

(*vi*) **Text To Speech scope :** The scope of this module is initiated with the conclusion of the receding module of Braille Recognition and Character Recognition. The module performs the task of conversion of the transformed Tamil text to audible format.

5. EXPERIMENT ANALYSIS

The phases of the above methodology are analysised by using the Tamil Braille System on Thirukkural Braille Book and Tamil Character Recognition on Thirukkural [27] text Book. Thirukkural [28] [29] is a classic Tamil literature consisting of 1330 Kurals or Couplets. 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. Following are the snapshots on result of the analysis.A *Kural* or *Couplet* consists of seven *cirs*, with four *cirs* on th e 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.* When embossed in Tamil Braille [30] format, if the space in not sufficient for the forth word in the first line, it is embossed in second line and the three *cirs* in the second line be embossed in the third line as shown in the Figure 21.

Fig. 17. Thirukkural in 3 lines.

1. Braille Image processing

Fig. 18. Braille image of 1st Kural

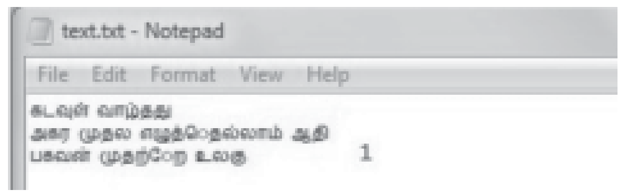

Fig. 19. Text conversion of the Figure 18

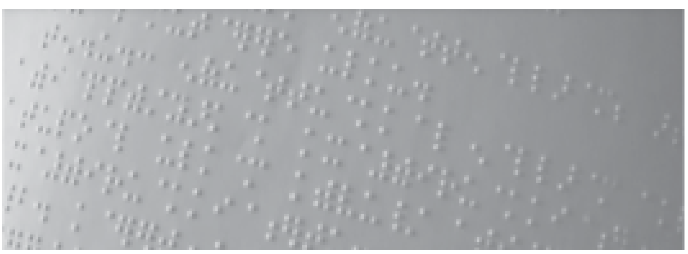

Fig. 20. Braille image of a tilted document

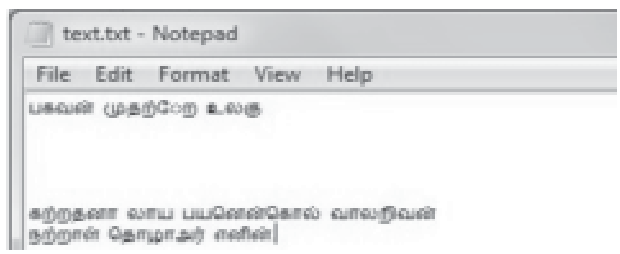

Fig 21. Text conversion of the Figure 20

2. Text Image processing

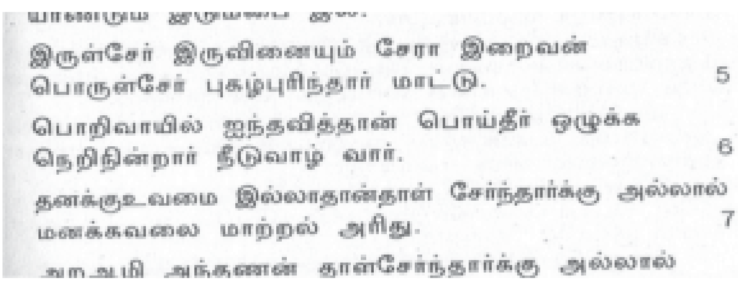

Fig. 22. image of Tamil Kural

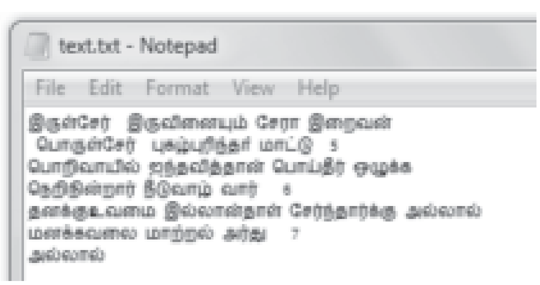

Fig. 23. Text conversion of the Figure 22

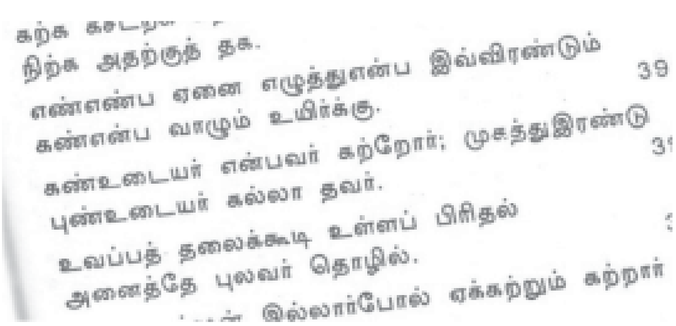

Fig. 24. Image of a tilted document

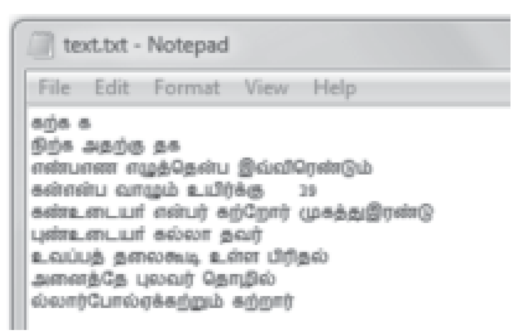

Fig. 25. Text conversion of the Figure 24

The post processing 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 Board.

6. CONCLUSION

Adaptive thresholding technique that has been used to separate the Braille dots from the background is an effective technique and it gives a very good result for more than 90% from the images. Braille has been developed as the reading and writing system for the visually impaired. The attention was given on this is very difficult to teach a visually impaired people in the early stage and more training is needed for teaching them and converting Braille to text, is costly and cumbersome work. The device implemented with the methodology detailed in this paper helps the visually challenged to read Tamil Text books and Braille documents. The present implementation is for the conversion of Braille and printed form of Tamil text. The device can be used like an image capture camera and captured over a document. The input image is taken by a camera in the hand-held [16] device and the output is given as speech through microphone or mini speaker using the above hardware interface. The software implementation and the hardware interfacing were done using Embedded Linux.

7. REFERENCE

1. Bharati Braille URL
2. http://acharya.iitm.ac.in/disabilities/bh_brl.php
3. Vandana Pushe, Nidhi Bhalla and Rupinderdeep Kaur, "Architecture of Gurmukhi to Braille conversion system", International Journal of Computer Science and Inormation Technology & Security(IJCSITS),Vol.2,No.2,April 2012, pp. 467-771
4. J.Bhattacharya, S.Majumder, G.Sanyal, "Automatic inspection of Braille Character: A vision based approach." International Journal of computer & Organization trends – Volume 1 Issue 3 – 2011.
5. Vandana Pushe, Rupinderdeep Kaur,"Issues in Converting Punjabi to Braille",Asian Journal of Computer Science and Information Technology 1:4,2011,pp.120-122.
6. Manzeet Singh, "Automated conversion of English and Hindi text to Braille representatoin". Intrenational Journal of Computer Applications – Volume 4 – No.6, July 2010.
7. Abdallah M.Abualkishik, and Khairuddin Omar "Quranic Baille System", World Academy of Science, Engineering and Technology,4-6,2008.
8. Mohammad Abu Yousu and S.M.Sai Shams, "Bangla Braille Inormation System: An Affordable System for the Sightless Population", Asian Journal of Information Technology 6 (6): 2007, pp.696-699.
9. Stanley, Paul B.; Karshmer, Arthur I.;Translating MathML into Nemeth Braille Code, Lecture Notes in Computer Science, Vol. 4061 LNCS, 2006, pp.1175-1182.
10. Al-Salman(2003), "Towards a Computerized Arabic Braille environment,PDF", King Saudi University.
11. Blenkhorn, Paul; "System For Converting Print Into Braille", IEEE Transactions on Rehabilitation Engineering, Vol. 5, No. 2, 1997, pp. 121-129.
12. Trent,S.D. &Truan, M.B (1997). "Speed, accuracy, and comprehension of adolescent braille readers in a speialized school". Journal of Visual Impairment and Blindness,91, pp.494-500.
13. K.H.Aparna, V.S.Chakravarthy (2003), "A Complete OCR System Development of Tamil Magazine Documents", Tamil Internet 2003, Chennai, pp. 45 – 51.
14. 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,p1297–1305.
15. 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.
16. 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.

17. 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.

18. Aparna K G and A G Ramakrishnan , "A Complete Tamil Optical Character Recognition System".

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

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

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

22. www.raspberrypi.org

23. Capt.Dr.S Santhosh Baboo, V.Ajantha Devi; "COMPUTER BASED TAMIL BRAILLE SYSTEM–A Review";11th International Tamil Internet Conference,Annamalai University, Chidambaram, India , Jointly conducted by International Forum for Information Technology (INFITT) and Annamalai University Dec 28-30, 2012. Pg.57 – 61

24. Capt.Dr.S Santhosh Baboo, V.Ajantha Devi; "TAMIL BRAILLE SYSTEM: A Conversion methodology of Tamil into Contracted Braille Script (Grade2)"; UACEE International Journal of Advances in Computer Science and its Applications – IJCSIA Volume 3: Issue 2 [ISSN 2250 – 3765] Publication Date: 05 June 2013 – pg.72 -76

25. Capt.Dr.S Santhosh Baboo, V.Ajantha Devi; "TAMIL BRAILLE SYSTEM: A Conversion methodology of Tamil into Contracted Braille Script (Grade2)"; Second Intl. Conf. on Advances in Computer, Electronics and Electrical Engineering -CEEE 2013, Institute of Research Engineers and Doctors. ISBN: 978-981-07-6260-5 doi: 10.3850/ 978-981-07-6260-5_36. pg171 – 175.

26. Capt.Dr.S Santhosh Baboo, V.Ajantha Devi; "Embedded Optical Character Recognition On Tamil Text Image using Raspberry Pi ";International Journal of Computer Science Trends and Technology (IJCST) – Volume 2 Issue 4, Jul – Aug 2014. pg 127 – 131.

27. Capt.Dr.S Santhosh Baboo, V.Ajantha Devi; "Embedded Optical Braille Recognition On Tamil Braille System using Raspberry Pi ";International Journal of Computer Technology and Application (IJCTA) - Volume 5 Issue 4, Jul–Aug2014.pg1566–1574.

28. https://en.wikipedia.org/wiki/Tirukku[a7.

29. www.thirukkural.com.

30. http://www.theiab.org/support-braille-books-production-house.html