Word Based Text Extraction Algorithm Implementation in Wearable Assistive Device for the Blind

Arunima B Krishna
Robotics Interest Group
Mechatronics/Robotics Laboratory
National Institute of Technology Calicut
Kerala, India 673601
Email: arunima8b8krishna1998@gmail.com

Meghana Hari
Robotics Interest Group
Mechatronics/Robotics Laboratory
National Institute of Technology Calicut
Kerala, India 673601
Email: meghanahari@gmail.com

Dr. Sudheer A.P
Robotics Interest Group
Mechatronics/Robotics Laboratory
National Institute of Technology Calicut
Kerala, India 673601
Email: apsudheer@nitc.ac.in

***Abstract*—The absence of proper Braille based resources for all kind of documents has adversely affected the life of Visually Impaired people. This paper presents a wearable assistive device for the blind which converts the text into acoustic output enabling the user to read any sort of text. The focus of this paper is the standalone Raspberry Pi based system with finger mounted camera that can help the visually impaired people in word based reading of the textual data pointed to by the finger. The system consists of a webcam that captures images which are enhanced. Following this the word pointed by the finger is extracted using a novel methodology and given to an Optical Character Recognition (OCR) engine. Subsequently, the textual output is given to a Text to Speech (TTS) converter to obtain audio via earphones.**

Keywords—Word-based Text extraction, Assistive Wearable Device, Finger Mounted Camera, Text To Speech (TTS)

I. INTRODUCTION

One of the major hindrances in the lives of visually impaired is the fact that they find it difficult to get resources to read. This is especially experienced when they visit shops, restaurants, hospitals etc. and have to depend on others. Various technologies have been developed to make life simpler for them. The initial implementations are more based on technologies to generate the Braille equivalent of the required text. Tsukuda et al. [1] have invented a Braille printer which embossed Braille characters for the input text given. This has its own practical difficulties which include the time constraint, bulkiness, cost etc. An instantaneous solution cannot be expected in this case. It is also not practical to get equivalent Braille script for all the texts. In this scenario, thoughts about other forms of technology began to arise, which could give other forms of feedback of whatever is printed in the text.

Creating acoustic versions of books was tried after Braille. Cassettes or tapes which have the book does not provide the reader a proper positioning once started. One cannot divert to the interested areas of the text. Hearing the text excerpts repeatedly is not practically possible or achieved. The Digital Accessible Information System (Daisy) standard describes an open data format for the representation of interactive books that are accessible to those with print-related disabilities. Daisy books may have both a textual and an audio component and allow for an active reading experience [2].

Other early assistive devices for the blind, where the visual signals of words are converted to nonverbal acoustic or tactile output include the Optophone by D'Albe [3], which uses musical chords or motifs and the Optacon (OPtical to TActile CONverter) by Goldish and Taylor [4], which uses a vibrotactile signal. Optacon requires the user to move the hand held camera while reading line by line. This provides difficulty in orienting the text properly and the inability to freely use both hands.

With advances in computation, the next form of innovation came in the form of extracting all the text present in an image which is the motivation behind Optical Character Recognition (OCR). Once an image is sent to the algorithm, it extracts all the text in that image. With the advent of smart phones and developments in OCR and Speech Synthesis, mobile based reading assistive application KNFB Reader, which converts the photo captured by the camera into text and reads it aloud [5]. Similarly Blindsights Text Detective also reads text from camera in phone. However it needs clear text placed at a particular distance from the camera for optimum results [6]. Such mobile based software thus require proper alignment, lighting and focus for accurate results and helps in converting a block of text in an image as a whole, which may not provide relevant information that the user intents to read.

Specialized devices such as the Eye-Pal, a battery powered portable and compact document reader for blind individuals, help in reading and analyzing texts [7]. Wearable devices have also been developed such as the OrCam, a commercial head-mounted camera system designed to recognize objects and read printed text in real time. Text-to-speech is activated by a pointing gesture in the cameras field of view [8]. Similar to this is iCare by Hedgepeth et *al.* [9], which scans the document using a Video Camera and extracts the text from the image. The text extracted is given as an auditory output through a small speaker which is attached to the device worn near the ear. The entire system is mounted on the table with a computer controlled by swivel and tilt mechanism. The device is quite bulky and it takes an overall picture of the page of the book opened. The bulkiness and a lack of real time effect remains the drawback of the work.

More closely related to this paper is Finger Reader, by Shilkrot et *al.* [10] works similar to the iCare. However it is a finger wearable device which extracts images of the text and outputs the word pointed to by the finger. The wearable set-up consists of the camera module and

978-1-5386-9543-2/19/$31.00

vibration motors which provide haptic feedback. The image taken by the camera is processed using the developed software which includes text-extraction algorithm, Tesseract OCR and Flite TTS in a standalone PC application. The finger mount device contains driver interface to connect to the PC. The device converts the line pointed by the finger into text, and words with high confidence are read aloud to the user. While skimming through the text user can hear multiple words near the finger. There are limitations to the portability of the system. A visually impaired person getting acquainted to the software interface is another practical difficulty. A similar device Handsight by Stearns et *al.* [11] which uses a 1x1mm 2 AWAIBA NanEye 2C camera connected to a wristband providing the wireless connectivity to PC on which the custom software runs. However the study has been conducted on an iPad rather than a physical prototype, and the experience of reading with an iPad and paper may be different.

Hence, as per the above literatures, the unavailability of a computationally inexpensive algorithm is identified. In addition the current systems implement software based technology utilising complex hardware. To the best of the authors' knowledge from the available literature it is found that there is no efficient algorithm to sort these issues. This paper hence puts forward a simple algorithm which gives satisfactory results with cost effective methodologies.

II. Design of the Wearable Assistive Device

The wearable assistive device operates in two modes, namely Offline mode and online mode. Operation in the offline mode requires a Web Camera to capture the image, which is sent to a single board computer, Raspberry Pi 3 Model A+ and finally the audio output delivered via earphones. The block diagram depicting the same is shown in Fig. 1. The device in the online mode functions almost in a similar manner except for the fact that the processing happens in a virtual Machine. Hence the image captured is sent to the Virtual Machine by Raspberry Pi which sends the text file with the text extracted back.

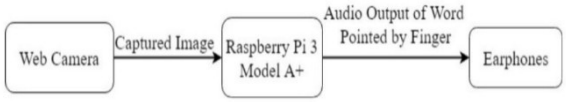

Fig.1 Block Diagram of Offline Mode

This is given as an audio output through earphones as evident from Fig. 2.

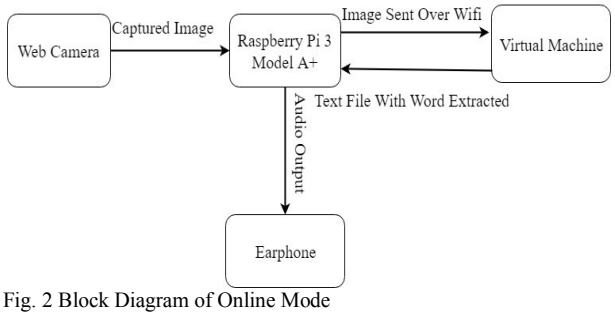

Fig. 2 Block Diagram of Online Mode

III. Methodology

This paper focuses on a standalone system with a finger mounted camera that supports the visually impaired in reading printed text by scanning with the finger mounted device. This work consists of novel preprocessing algorithms and different process modalities. The finger-worn camera helps in obtaining focused images at a fixed distance and helps the user to utilize the sense of touch when scanning the surface of the document.

AHardware

The device consists of an index mounted webcam connected with Raspberry Pi 3 model A+. The image captured by the camera is processed by the computer. The word pointed by the finger is provided as an acoustic output through the earphones plugged in to the module.

BSoftware

The system deals with the following main processes

- Image acquisition
- Pre-processing the image
- Word based text extraction

It also incorporates Integration layer with Tesseract OCR and Espeak. The data flow diagram is given in Fig. 3

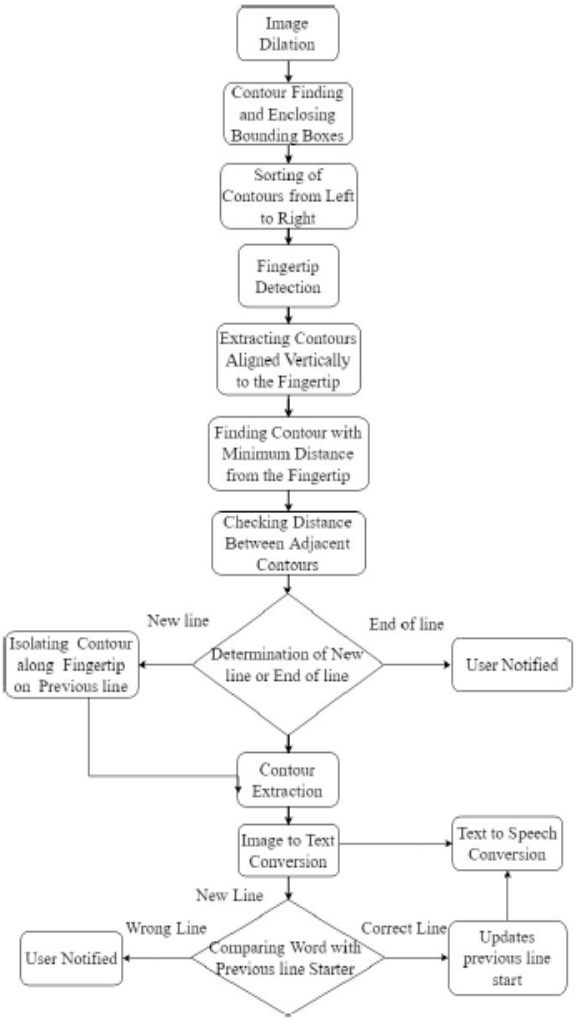

Fig. 3 Data Flow Diagram

Word Extraction Algorithm: The basic steps involved in preprocessing as shown in Fig. 4 are :

- Image Enhancement
- Exponential Transformation
- Skew Correction

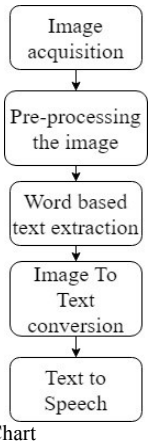

Fig. 4 Preprocessing Flow Chart

The word extraction procedure and the user-feedback methodology shown in Fig. 5 includes the following:

- Boundary boxes around words
- Fingertip detection
- Word Isolation

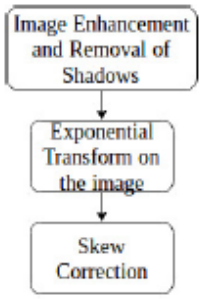

Fig. 5 Flow Chart of the Word Extraction Algorithm

Image preprocessing is done using Open CV, an open source computer vision library under the Berkeley Software Distribution (BSD) open source license.

The image captured is re-sized to 800x600 so that the subsequent procedures act on a uniformly sized images and give similar results for all inputs.

- *Image Enhancement:*

The image in RGB format is split into individual channels and it is dilated to blur the text present. Median blur is applied on the result with a decent sized kernel to further suppress any text. This gives a background image that contains all the shadows and blobs. The difference between the original and the background is obtained. The bits that are identical are black (with less difference), and the text will be white (large difference). The result is inverted to get black text on white image. The image is normalized to obtain an image with full dynamic range. Other methods of image enhancement such as histogram equalization, Contrast Limited Adaptive Histogram Equalization (CLAHE), brightness and contrast adjustment using gain and bias parameters, gamma correction have also been implemented. However they do not provide uniform outputs for different kind of images taken under different lighting conditions.

- *Exponential transform:*

In order to obtain images with distinct edges of the letters, an anti logarithmic transform is applied. It has provided better results than compared to applying sharpening filters on the image

If 'r' denotes each pixel in the input image, 's' each pixel in the resulting image and 'c' be a constant, then they are all related by the expression

$$s = ce^{(r-1)} \quad (1)$$

The pixel values are scaled and exponential transform is done on them. The pixels are inverted to obtain the image with white text on black background. The motivation behind this operation is evident from the curve in Fig. 6, where the exponential transform becomes steep at the other end of the

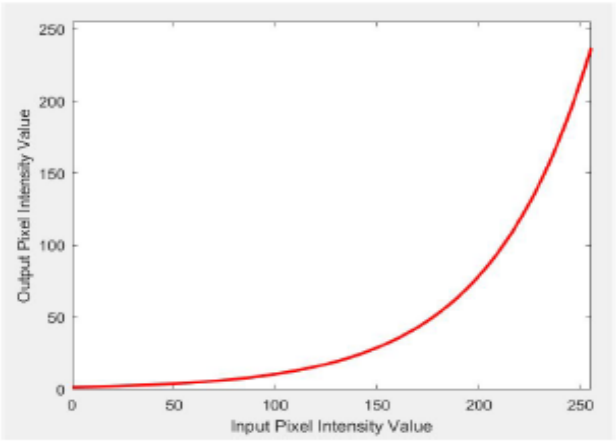

Fig. 6 Curve of Exponential Transform

spectrum, thereby mapping a narrow range of input values to a much larger range at the output. The minimum and maximum values of the input and output pixels remain 0 and 255 respectively after the transform. The entire image is darker than the original gray scale band in the image.

- *Skew Correction:*

The image is binarized and then dilated with a kernel of size 7x7. The kernel size is selected based on the font size of the text in the image. The contours are found for the dilated image and minimum area rectangle is extracted for the contours which have width in the range of 20 - 300 pixels and height in between 20 and 150. This is done to avoid the blobs or other non-text objects from being detected. The angle of rotation of the rectangles are estimated and average value is calculated. The image is rotated by the obtained angle. Further operations are done on the rotated image.

- *Bounding boxes around words:*

The image is again dilated so that the contours are correctly obtained for the words, for which bounding boxes are drawn on binary image of same size. The coordinates of the rectangles are taken and they are sorted in the order from left to right.

- *Fingertip Detection*

The Red Green Blue (RGB) image is converted to Hue Saturation Value (HSV) color model. The image is thresholded such that the skin color can be isolated. The entire skin region is taken as a contour and its extreme coordinate is taken as its finger tip.

- *Word Isolation:*

The moments of all the contours which have the same abscissa as that of the fingertip are calculated. The distance between the moment and the fingertip point is calculated for each contour and the minimum distance is found. The corresponding contour is cropped out from the image and placed on normal white image similar in size as that of input image.

- *Image To Text conversion:*

The final image is given to the Tesseract OCR Engine [12], which generates a text file. The special characters in

the text file are removed and then given to the TTS converter.

- *Text to Speech:*

eSpeak [13], an open source software speech synthesizer is used to convert the text file to audio output which is heard through the earphones. Python 3 wrapper for eSpeak is utilized to implement the algorithm.

The user is informed about placement of their hand on the text document such as the beginning of a line, end of a line, deviation from line, positioning at the correct line after an end of line etc. with an acoustic feedback mechanism.

- *The beginning of the line :*

The index of the contour pointed by the finger is taken and also the adjacent contours are extracted. The horizontal and vertical distances between each of these are calculated. If the vertical distance between the next contour and the current one is less than the distance between the previous and the current one, then the word pointed is treated as the beginning of a new line.

- *End of line:*

End of a line is detected when the the distance between the current and next contour is more than that between previous and current one.

- *Deviation from the line:*

The distance between the finger tip and the contour adjacent is kept in between a range of values. Deviation form this is monitored in each image and the user is notified whether to move finger upwards or downwards.

- *Finger position correction:*

The text obtained at the beginning of the line of a sentence is stored and then compared with the text extracted from the contour above the first word of the next line. The line on which the finger is present is correct if the comparison yields zero.

Another approach is to use cloud based processor. The image taken for from the webcam is sent to cloud based virtual machine using Secure Copy Protocol, which uses Secure Shell (SSH) for data transfer. The text file generated is sent back to the Raspberry pi and eSpeak module is used to convert to audio output.

IV. Experimentation And Trials

The camera set up is as shown in Fig. 7 and the image captured by the camera is shown in Fig. 8.The image is enhanced and shadows removed as shown in Fig. 9. After enhancement, exponential transform is done on the image. The words are distinctly visible in resulting image as shown in Fig. 10.

Once the clear image is obtained, it is de-skewed as given in Fig. 11 so that it can be accurately converted by the OCR Engine. Once the corrected image is obtained the individual words have to be located. Thus contours are found on the image after dilation. Bounding boxes are drawn on each of the words as shown in Fig. 12. The fingertip is detected by converting the image into HSV model and extracting the extreme point on the isolated contour of the finger. The line from the fingertip to the end of the image is drawn as evident from the Fig. 13. The coordinates of the boxes obtained and the fingertip are compared. The contour closer to the fingertip is extracted as shown in Fig. 14. This is provided to the OCR Engine for converting it to text.

Fig. 7 The Camera Setup

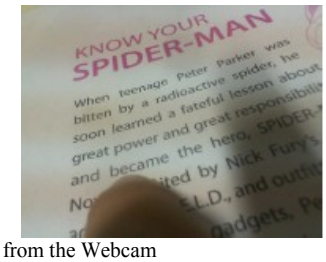

Fig. 8 Image taken from the Webcam

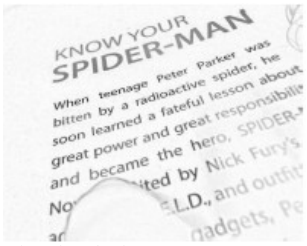

Fig.9 Image after Removing the Shadow

Fig. 10 On applying Exponential Transform

V. Results And Discussions

The final image that is given to OCR Engine as shown in Fig. 14 is 800 X 600 in size, and the word isolated is of the same size as that in the input image. The algorithm is implemented in Raspberry Pi 3 Model A+. The entire process from capturing the image, processing to extract the intended word and final audio output to the earphone takes approximately 5s. Raspberry Pi 3 Model A+ has a RAM capacity of around 512MB with a 1.4GHz clock speed.

The algorithm is also implemented on a Virtual Machine (VM) provided by Cloud Computing facility of Microsoft named Azure Cloud. In this, the subscription is for a Virtual Machine of 32GB RAM. The image is captured in Raspberry Pi Model A+ and sent to the VM using the Secure Copy Protocol. The processing is done in the VM and the extracted word is sent to the Raspberry Pi in the form of a .txt file. The final conversion of the .txt file to audio using Python eSpeak Text to Speech Converter Module is done in Raspberry Pi and the output delivered through an earphone. The entire process takes approximately 0.5s and the approximate accuracy of the word to audio conversion was found to be 75% for a black on white document. The special characters obtained in the text file after image to text conversion is removed to prevent error in TTS conversion

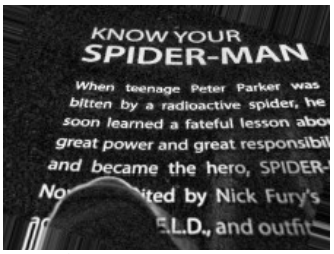

Fig. 11 After Skew Correction

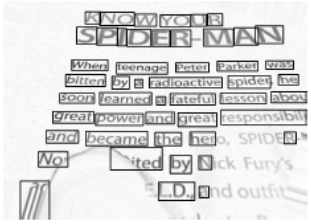

Fig. 12 Bounding Boxes for Each Word

VI. Conclusion And Outlook

The proposed wearable assistive device provides simple and effective assistance to the visually impaired by aiding them to read any text available with them using an efficient algorithm. The model possesses various advantages such as portability, computational effectiveness and affordability when compared to the other existing technologies. Despite this, various shortcomings of this technology includes the requirement of uninterrupted fast Wi-fi connectivity, sensitivity to varying lighting conditions and added expense to maintain a cloud facility. This technology can be further enhanced by developing Machine Learning algorithms to implement OCR for adversely lit image inputs. Optimizing the available algorithm to make the device standalone without the aid of a Virtual Machine also provides a possibility for future extension.

Fig. 13 Fingertip Detection

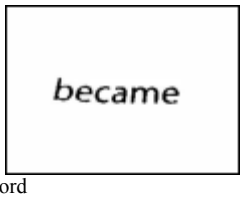

Fig. 14 The Extracted Word

References

[1] Y. Tsukuda, I. Goto, "Braille Printer", US5193921A,(1991)

[2] D. Leas, E. Persoon, N. Soiffer, and M. Zacherle, Daisy 3: A Standard for Accessible Multimedia Books, IEEE Multimedia, vol. 15, no. 4, pp.2837, Oct. 2008.

[3] E. E. F. DALBE, The Optophone: An Instrument for Reading by Ear, Nature, vol. 105, no. 2636, pp. 295296, Jun. 1920.

[4] L. H. Goldish and H. E. Taylor, "The optacon: A valuable device forblind persons," New Outlook for the Blind, Nov. 1973.

[5] Knfbreader.com. (2010).KNFB Reader App features the best OCR. Turnprint into speech or Braille instantly. iOS 3 now available. — KNFB Reader. [online] Available at: https://knfbreader.com/ [Accessed 15 Feb.2019].

[6] Blind Help Project. (2013). Text Detective. [online] Available at: http://blindhelp.net/software/text-detective [Accessed 15 Feb. 2019].

[7] ROL, E. (2013). Abisee Eye-Pal ROL. [online] Woodlake Technologies, Inc. Available at: http://www.woodlaketechnologies.com/Eye-Pal-ROL-p/abi800.htm [Accessed 21 Feb. 2019].

[8] OrCam. (2013). Help Peoplewho are Blind or Partially Sighted - OrCam. [online] Available at: https://www.orcam.com/en/ [Accessed 6Feb. 2019].

[9] T. Hedgpeth, J. A. Black, Jr., and S. Panchanathan, A demonstration of the iCARE portable reader, Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility - Assets 06,2006.

[10] R. Shilkrot, J. Huber, C. Liu, P. Maes, and S. C. Nanayakkara, Finger-Reader, Proceedings of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems - CHI EA 14, 2014.

[11] L. Stearns, R. Du, U. Oh, Y. Wang, L. Findlater, R. Chellappa, and J. E. Froehlich, The Design and Preliminary Evaluation of a Finger-Mounted Camera and Feedback System to Enable Reading of Printed Text for the Blind, Lecture Notes in Computer Science, pp. 615631, 2015.

[12] GitHub. (2016). tesseract-ocr. [online] Available at: https://github.com/tesseract-ocr/ [Accessed 26 Jun. 2018].

[13] Espeak.sourceforge.net. eSpeak Speech Synthesizer. [online] Available at: http://espeak.sourceforge.net/docindex.html [Accessed 26 Jun. 2018].