Image Text to Speech Conversion using Optical Character Recognition Technique in Raspberry PI

Mangesh Sarak[1], Prof. S. S. Patil[2], and Prof. Abhijit S. Mali[3]
[1]Student, Department of Electronics and Telecommunication Engineering, Tatyasaheb Kore Institute of Engineering and Technology (An Autonomous Institute), Warananagar, Kolhapur, 416113, Maharashtra, INDIA
[2]Professor, Department of Electronics and Telecommunication Engineering, Tatyasaheb Kore Institute of Engineering and Technology (An Autonomous Institute), Warananagar, Kolhapur, 416113, Maharashtra, INDIA
[3]Professor, Department of Electronics and Telecommunication Engineering, Tatyasaheb Kore Institute of Engineering and Technology (An Autonomous Institute), Warananagar, Kolhapur, 416113, Maharashtra, INDIA

[1]Corresponding Author: mangeshsarak@gmail.com

Received: 21-05-2024 Revised: 09-06-2024 Accepted: 29-06-2024

ABSTRACT

Optical Character Recognition (OCR) is a subset of artificial intelligence and is a subset of computer vision. Optical Character Recognition (OCR) is the use of Raspberry Pi to convert scanned bitmap images of handwritten or written text into audio performance. OCRs designed for a variety of world languages are now in use. In this method the context subtraction method based on the Gaussian mixture is used to recover the area of the moving object. For text content, the function of text localization and recognition is used. The text localization algorithm and the Tesract algorithm and edge pixel distributions based on the gradient properties of the stroke directions were used to automatically translate text areas from the object in the Ada enhancement model. In the translated text areas text characters are converted to binaries, which OCR software understands. For the blind, known text symbols are strongly pronounced. The potential of the algorithm for the proposed text location. The text file describes the character codes using the Raspberry system, which recognises the characters by using Tesract's and Python, and the audio output is heard in the recognition step.

***Keywords*—** Image, Text, Speech, PI

I. INTRODUCTION

Of the 7.4 billion people on our globe, 285 million are blind or visually handicapped million individuals are totally blind. i.e. have no vision at all and 246 million have mild or severe visual impairment (WHO, 2011). It is projected that by 2020, there will be 200 million individuals with vision impairment and 75 million blind persons [5]. As reading is of prime importance in the daily routine (text being present everywhere from newspapers, commercial products, sign-boards, digital screens etc.) of mankind, visually impaired people face a lot of difficulties. By reading the material aloud to the visually handicapped, our gadget helps them. Many developments in this field have made it easier for those who are visually impaired to read without much difficulty. The existing technologies use a similar approach as mentioned in this paper, but they have certain drawbacks. First off, the test inputs are printed on a plain white sheet; the input photos from earlier efforts don't have any complicated backgrounds. It is easy to convert such images to text without pre-processing, but such an approach will not be useful in a real-time system [1][2][3]. Additionally, characters that are recognized using segmentation will be read out as individual letters rather than as whole words in these ways. The user hears an unwanted auditory output as a result. For our project, we needed the gadget to be able to efficiently read the text even against complex backgrounds. Motivated by the approach taken by applications like "CamScanner," we reasoned that the text would probably be encased in a box on any complicated background, such as billboards, displays, etc. We assume that the region containing four points is the necessary region holding the text because we can detect it. Cropping and warping are used for this. After undergoing edge detection on the newly acquired image, a boundary is drawn across the letters. It gains additional definition as a result. After that, the image is processed using OCR and TTS to produce audio.

II. LITERATURE REVIEW

Reduced clarity of vision that cannot be corrected with glasses is known as visual impairment or vision loss. Complete visual loss is referred to as blindness. Cataracts, glaucoma, and uncorrected refractive errors are the common causes of vision loss. Individuals who are visually impaired encounter several challenges when engaging in routine tasks such as driving, walking, and reading. [6]

Braille

People with vision impairments utilize the Braille writing and reading method. Braille is written on paper that has been embossed. The braille characters are discrete, rectangular blocks known as cells that have raised dots, or bumps, on them. The raised dots are arranged such that the visually impaired person may feel the information being conveyed. [7]

While braille readers, keyboards, and monitors are available, they are not easily accessible to remote people, and there is a lack of readily available braille content.[8]

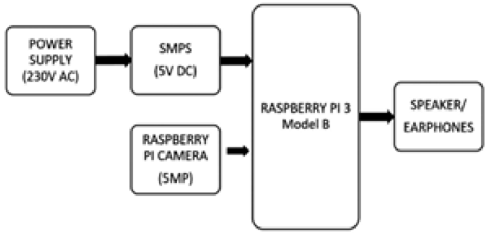
Raspberry pi

With the help of a keyboard, mouse, and display, the inexpensive Raspberry Pi CPU may be upgraded to a powerful, full-featured computer [9]. We choose the Raspberry Pi microcomputer for our project because, in the first place, it is an inexpensive, widely accessible gadget. The fact that the software used by Raspberry Pi is either open source or free adds to its affordability. The Raspberry Pi has the benefit of portability due to its compact size and uses an SD card for storage[10]. The Open CV (Open source Computer Vision) libraries are used for image processing during the software development process. The image processing coder was taken into consideration while designing each function and data structure[11]. Current systems and their constraints. The portability of barcode readers is one of their main benefits. As a result, the blind can utilize them to distinguish between various products. All of the product's information is compiled into one large database. With e-braille readers, the user only needs to scan the bar code to view the product data. One potential drawback of this device is that it may be difficult for the user to direct the bar code reader in the right direction. [2] Optical enhancement options, such an optical zooming device that enlarges the braille character, are another strategy. But not everyone who is blind or visually handicapped needs to be able to read in braille. [4]. A few techniques try to turn text into speech. A computer, speakers, and a scanner are used to achieve this. This technique works best with basic scanned materials. It can't extract text from a picture where the background is complicated. [4]

III. PROPOSED SYSTEM

SYSTEM OVERVIEW

Block diagram



Hardware Specifications

Raspberry pi

The Raspberry Pi is a gadget that has multiple vital features on a single chip. It's a SoC, or system on a chip. The Broadcom BCM2837 SoC Multimedia processor powers the Raspberry Pi 3. The 4x ARM Cortex-A53, 1.2GHz CPU powers the Raspberry Pi. It includes a 1 GB LPDDR RAM (900 Mhz) internal memory and up to 64 GB of expandable external memory. The Raspberry Pi 3 boasts two primary new features: Bluetooth 4.1 classic and 802.11n wireless internet connectivity. 40 GPIO pins are present. The resolution of the 5MP Raspberry Pi camera is 2592 x 1944. The Raspberry Pi features a 3.5mm audio connection, which makes it simple to connect speakers or earbuds to it in order to hear audio.

Figure 2: Schematic diagram of Raspberry pi

Camera Module

A flat flex cable (FFC, 1mm pitch, 15 conductors, type B) is used to connect the Raspberry Pi camera module, which has a 5MP sensor and measures 25 mm square. The Raspberry Pi computer is connected to the camera module.

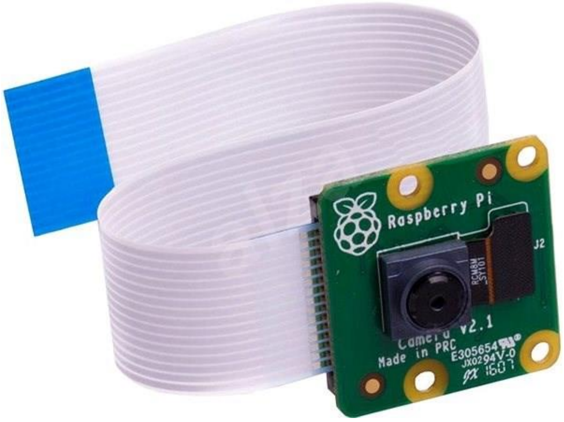


Figure 3: Raspberry Pi Camera Module

With the following crucial features, the Raspberry Pi camera module offers a novel new capability for optical instrumentation:

1080p video capture stored on SD flash cards. HDMI simultaneous output of 1080p live video while recording is possible. Sensor type: Omni Vision OV5647, 5 megapixel CMOS QSXGA color sensor 3.67 x 2.74 mm for the sensor, 2592 x 1944 for the pixel count, 1.4 x 1.4 um for the pixel size, and f=3.6 mm and f/2.9 for the lensField of View: 2.0 x 1.33 m at 2 m, Angle of View: 54 x 41 degrees 35 mm equivalent full-frame SLR lens with fixed focus from 1 m to infinity Lens that is removable. Lens interchanges adapters for M12, C-mount, Nikon F Mount, and Canon EF. Mirroring images in-camera.

Software Specifications

Based on Debian, Raspbian is a free operating system designed specifically for the Raspberry Pi computer. The primary operating system for the Raspberry Pi in our project is Raspbian Jessie. The Python (2.7.13) programming language is used in our code, and OpenCV is used to call the functions. OpenCV, an acronym for Open Source Computer Vision, is a collection of functions utilized for real-time applications such as image processing, among numerous more uses [14]. As of right now, OpenCV is accessible for several operating systems and programming languages, such as Windows, Linux, OS X, Android, iOS, and C++, Python, and Java, among others. OpenCV-3.0.0 is the version we are using for our project. Among the various application areas of OpenCV are gesture recognition, augmented reality, mobile robots, motion understanding, object identification, segmentation and recognition, facial recognition systems, motion tracking, and human–computer interaction (HCI). Tesseract OCR and Festival software are installed in order to perform TTS and OCR operations. The Apache 2.0 license can be used to access Tesseract, an open-source optical character recognition (OCR) engine. It can be used to extract printed, handwritten, or typed text from photos directly, or (for programmers) via an API. An extensive range of languages are supported. 'Tesseract' or 'tesseract-ocr' is the common name for the package. The UK's "The Centre for Speech Technology Research" is the organization behind Festival TTS. An effective framework for creating speech synthesis systems is included in this open-source program. Supports Spanish, American English, and British English. It is multilingual. Installing Festival is simple because it comes with the Raspberry Pi package manager.

Image Processing

Letters are found in books and documents. Our goal is to extract these letters, digitize them, and then repeat them appropriately.

The letters are obtained using image processing. In essence, image processing is the application of a set of operations on an image file in order to extract certain information from it. An image serves as the input, while an image or a set of parameters derived from the image can serve as the output. We can transform the image to a grayscale image once it has been loaded.

We now have an image in the form of pixels within a predefined range. The letters are determined using this range. The image is either black or white in grayscale; the white will primarily consist of blank space or the spaces between text.

Feature Extraction

We compile the image's key characteristics in this step, known as feature maps. Finding the image's boundaries is one way to do this because those areas will have the necessary text on them. Several axes detecting methods, including as Sobel, Kirsch, Canny, Prewitt, and others, can be used for this. The Kirsch detector is the most precise at locating the horizontal, vertical, right diagonal, and left diagonal axes. The eight point neighborhood of every pixel is used in this technique.

Optical Character Recognition

Text that has been handwritten, typewritten, or printed can be scanned and then mechanically or electronically converted into machine-encoded text using optical character recognition, or OCR for short. Data entry from original paper sources, such as documents, sales receipts, letters, or any other printed record, is done using this method on a large scale. It is essential to computerizing printed texts so that they can be utilized in machine processes like text-to-speech, machine translation, and text mining as well as electronically searched, stored more compactly, and shown online. OCR is a branch of computer vision, artificial intelligence, and pattern recognition research.

Tesseract

A free optical character recognition engine available for many operating systems is called Tesseract. One of the most precise free software OCR engines out there right now is Tesseract. It works with Windows, Mac OS, and Linux. The Tesseract engine is a command-based tool that receives a picture including text as input. The Tesseract command then processes it. The Tesseract command requires two parameters. The name of the text-containing image file is the first argument, and the text extraction's output file is the second. There is no need to provide the file extension when passing the output file name as a second argument in the Tesseract command because Tesseract automatically sets the output file extension to.txt. The output's content can be found in the.txt file once processing is finished. When working with basic grayscale photos, Tesseract yields 100% accurate results. However, Tesseract yields more accurate results for some complicated images when the images are in grayscale mode as opposed to color. Tesseract is a command-based tool, but it may be readily made available in graphical mode because it is open source and available as the Dynamic Link Library.

Software Implementation

System software: Raspbian (Debian)
Spoken language: Python 2.7
Platform: OpenCV (Linux-library), Tesseract OCR and TTS engines in the library
Derived from the Debian operating system, Raspbian is the operating system used for the proposed project.
Python is a scripting language that is used to write the algorithms. The OpenCV Library functions are invoked by the algorithm. Written in C and C++, OpenCV is an open-source computer vision library that works with Windows, Linux, and Mac OS X.

OpenCV was created with a significant emphasis on real-time applications and processing performance. Because OpenCV is designed in efficient C, it can run on multi-core computers.

IV. FLOW OF PROCESS

1. Image Capturing

The initial stage involves positioning the document beneath the camera and having it photographed by the device. Because of the high-resolution camera, the image quality will be excellent for quick and clear recognition.

2. Pre-Processing

Three phases make up the pre-processing stage: linearization, noise removal, and skew correction. We examine the obtained image for skewing. Either a left or a right orientation could cause the image to become distorted. Here, the picture is initially made brighter and more balanced.

The skew detection function looks for an orientation angle of between ±15 degrees. If one is found, the image is simply rotated till the lines align with the real horizontal axis, creating a skew corrected image. Prior to further processing, any noise that was added during capturing or as a result of the low quality of the page must be removed.

Here, various methods are used to remove the background from the input image. The image is then grayscaled, binarized (total black and white), and stored in a matrix of values. Utilizing the subsequent procedures, preprocess an image with stage noise.

Binarisation

• The load input image may be a BMP or JPG.
• Obtain and compute the n channels, height, and breadth.
• Obtain the pointer to the picture data.
• Using the manual r, g, and b channels, convert each height and width of the image to grayscale.
• Use the following method to convert an input RGB color image to a grayscale image Y.
$Y = 0.299R + 0.587G + 0.114B$
• Convert this greyscale image Y to a binary image (bitmap) in which the characteristic function is used to acquire the pixel values as indicated by b.
If $g(x, y) < T$ 25 = 0 and if $g(x, y) > T$
T= Intensity mean (Threshold),
then $B(x, y) = 1$.

Remove Line: Line removal in both the horizontal and vertical directions Using column and row as a guide, move the image's black pixels both vertically and horizontally. Then, all pixels should be turned white if the total number of pixels is less than 85% of the image's height and breadth.

Gap Removal: Once the vertical and horizontal lines are eliminated, an image discontinuity or gap will appear. Utilizing the 4-Connected Component algorithm to eliminate it. If a white pixel has at least one black pixel, it will detect that pixel and turn it black. With its support, we can effectively eliminate any gaps that may have formed in the image, aiding in the character recognition of discontinuous characters.

Segmentation

This step attempts to segment the input image using various region-based and connected component segmentation algorithms.

Region-Based Segmentation:

Text regions differ from non-text regions in terms of gradient distribution, texture, and structure, which is the foundation of region-based segmentation techniques [3]. Text detection and text localization are the two processes that these approaches typically include. In order to identify texts in a given local region, its features are extracted. Then, to precisely localize text sections, particular grouping or clustering techniques are used.

Connected Component Based Segmentation

Methods based on connected components (CC) [3][4] are based on the idea that texts can be viewed as collections of discrete connected components, with distinct enclosed outlines, colors, and intensities. Typically, these techniques consist of three steps: Three steps are involved in extracting CCs from images, analyzing CCs using classifiers or heuristic rules to decide if they are text components, and grouping text components into text regions like words, lines, etc. Even though some current approaches have shown encouraging outcomes, there are still a number of issues that are challenging to resolve. Text components are difficult to precisely segment using CC-based techniques in the absence of prior knowledge on text position and scale. Furthermore, because there are too many text-like components in photographs, it is challenging to create a quick and trustworthy CC analysis tool. However, the text orientation and cluster number have an impact on the region-based approaches' performance. The majority of these techniques are limited to localizing texts with a large character set in a horizontal alignment.

Character or Text Recognition/Detection

After the text in the image has been divided, it is utilized to identify each alphabet (letter). Every character is identified by use of the classifier. Character Matching verifies that each database template matches the input character segment by comparing the segments to characters in the database.

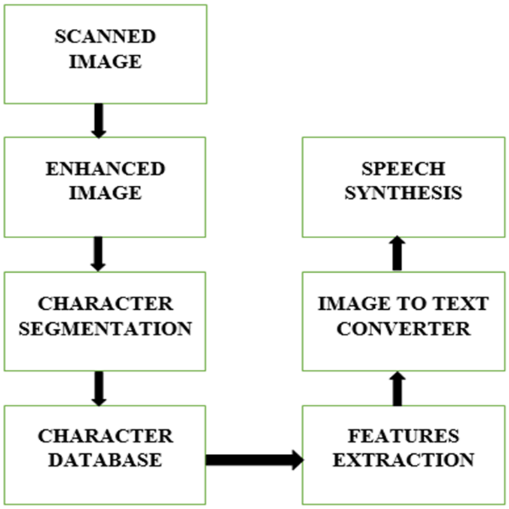


Figure 4: Flow of Process

V. IMAGE TO TEXT CONVERTER

The Raspberry Pi board processes the identified characters' ASCII values. In this case, every character is compared to its matching template and recorded as a transcription of normalized text. The audio output receives an additional delivery of this transcription.

Text to Speech

At the end of the receding Character Recognition module, the scope of this module begins. The work of turning the converted text into an auditory format is completed by the module. There is an on-board audio jack on the Raspberry Pi. The audio is produced using a PWM output and is only slightly filtered. The volume and sound quality can be significantly increased with a USB audio card.

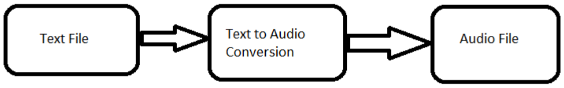


Figure 5: Block Diagram Text to speech

When the recognition process is finished, the audio output is played back. The character codes in the text file are processed using a Raspberry Pi device, which uses the Tesseract algorithm and Python programming to recognize a character.

Objectives

1. The first of our project's goals is to extract textual material, digitize it, and then recite it appropriately.
2. To serve as a successful communication tool.
3. To give visually impaired children a high-quality education.
4. Creating and designing a learning tool that provides a setting for independent study.
5. Create a playful environment while learning Braille writing in an incredibly economical way.

Methodology

Image Acquisition: The text images are taken in this stage by the built-in camera. The camera being utilized determines the quality of the image that is captured. We are using the 5MP, 2592 x 1944 resolution camera that comes with the Raspberry Pi.handling: This process includes thresholding, warping, cropping, noise reduction, edge detection, and color to grayscale conversion. Since many OpenCV functions require the input argument to be an image in grayscale, the image is transformed to that format. Bilateral filters are used to reduce noise. To improve contour detection, the grayscale image is subjected to canny edge detection. The image is cropped and twisted in accordance with its contours. This helps us eliminate the undesirable backdrop and identify and extract only the text-containing section. Thresholding is ultimately applied to the image to make it resemble a scanned document. This is done so that the image may be efficiently converted to text using OCR.

Image to Text Conversion: Figure (above) illustrates how text-to-speech works. The OCR and picture pre-processing components make up the first block. The preprocessed image, which is in.png format, is changed to a.txt file. Our tool of choice is Tesseract OCR.

Text to Speech Conversion: The voice processing module is located in the second block. The.txt file is converted to an audio output. Here, a speech synthesizer known as Festival TTS is used to turn the text into speech. The Raspberry Pi is equipped with an on-board audio jack, which is powered by a PWM output.

Algorithm

```
Step 1: Begin with the starting points
Step 2: Import sub-process and set up GPIO pins
Step 3: If the button is hit,
    i. take a picture using a webcam
    ii. Carry out Tesseract OCR
    iii. Save with thresholding into a text document
    iv. Festival software for text-to-speech operation.
Step 4: Repeat Step 3
```

Flowchart

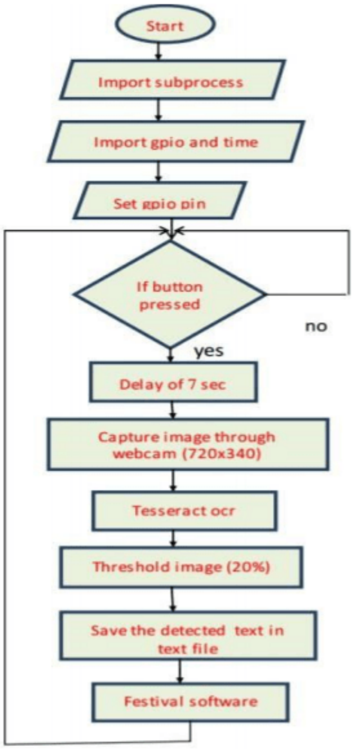


fig 6:-flowchart

VI. ADVANTAGES

- Improved word recognition skills, fluency and accuracy.
- Global market penetration.
- It is very reliable and user friendly.
- Enhanced employee performance.
- Extend the reach of your content.
- People with different learning styles.
- It is very useful for illiterates.

Application

- It is primarily intended for blind individuals.
- It translates text into spoken voice output.
- It aids in word identification and helps small types retain information while reading.
- For people who are illiterate, we can provide access to many languages.
- Text to Speech's Effect on Travel.
- It can be utilized as an announcement tool at colleges and schools as well as other locations.

VII. CONCLUSION

This method makes it simple for visually impaired people to listen to anything they wish to. Additionally, they can translate the text into the desired language with the aid of translation tools, and they can translate that modified text into voice again by utilizing Google's speech recognition engine. They can be independent in this way. Furthermore, it is less expensive than alternative implementations. With an average processing time of less than three minutes for A4 paper size, a text-to-speech device may convert text image input into sound with a performance that is high enough and a readability tolerance of less than 2%. People can utilize this portable device on their own and without an internet connection. We can simplify the process of modifying books or web pages by using this strategy.

REFERENCES

[1] Priyanka Muchhadiya. *The different image processing techniques to text from natural images*.

[2] Sanjay Dutta, Sonu Dutta, Om Gupta, Shraddha Lone & Prof. Suvarna Phule. *PISEE: raspberry pi-based image to speech system for the visually impaired with blur detection*.

[3] Prof. Vaibhav V. Mainkar, Miss. Tejashree U. Bagayatkar, Mr. Siddhesh K. Shetye, Mr. Hrushikesh R. Tamhankar, Mr. Rahul G. Jadhav & Mr. Rahul S. Tendolkar. *Raspberry pi based Intelligent Reader for Visually Impaired Persons*.

[4] Prashant Chougule. Raspberry Pi based reader for blind people. *International Research Journal of Engineering and Technology(IRJET)*.

[5] Sahana K Adyanthaya. Text recognition from images: A study. *International Journal of Engineering Research & Technology (IJERT)*.

[6] Dr. Merry-Noel Chamberlain. *Braille basics plus, transition booklet into unified English Braille (UEB)*. TVI, NOMC Edited by Faye Miller, MA, TVI, COMS.

[7] E – Braille. A study aid for visual impaired. *Chaitanya Jambotkar Jain College of Engineering & Research*.

[8] Anand Nayyar. *Raspberry Pi-A small, powerful, cost effective and efficient form factor computer: A review*.

[9] Hirak Ghael. A review paper on raspberry pi and its applications. *BK Birla Institute of Engineering and Technology, Pilani*.

[10] Ishita Pal, Mohammadraza Rajani, Anusha Poojary & Priyanka Prasad. *Implementation of Image to Text Conversion using Android App*.

[11] Qixiang Ye & David Doermann. *Text detection and recognition in imagery: A survey*.