Palestine Polytechnic University

College of Information Technology and Computer Engineering

Book Reader and Page Turner Robot using Raspberry pi

Submitted by:

Lana Sharbati Manar Iqefan

Omar Taradeh

Supervisor:

Dr. Mousa Farajallah

January,2021

Acknowledgement

In the name of "Allah "the most beneficent and merciful, who gave us strength and helpful us to get through this project.

We sincerely, thank our parents for their kind supporting to reach the end of point in our university life.

We would like to thank Dr. Mousa Farajallah for his support and assistance to complete this project.

Last but not least, we would like to thank our friends who went together step by step and hand by hand during the college years, and thanks for all teachers for giving us enough information, we are also thankful to everyone who supported us through many helpful comments.

Abstract

People who are unable to read texts, like people suffering from visual impairments, need to get new information from books and other written sources for educational purposes and entertainment. Although there are some audiobooks that they can listen to, there is still an abundance of books that they might need, which are hard to digitize mainly because of copyright issues. Therefore, there is a real need for using technology to help facilitate the reading process of a wide range of books and written material to help this group of people to overcome their reading difficulties.

To solve this problem, a robot that can read and turn the pages of the book is proposed in this project. The reading process is carried by capturing the book pages and extracting the text from the captured image. Then, the text is converted into a speech which can be heard via a speaker. After finishing reading each page, the robot will turn the page using a wheel and arm, using different software and hardware components.

The reading robot described above is expected to read specific types of books and, thus, help various people who have reading difficulties gain access to the informational content and entertaining content included in books and written material.

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview of the project

Technology contributes to solving various human problems. Robots, for example, are independent machines that can make decisions and do procedures. Robots have many types. Movable robots, for instance, can move in a certain range, and stationary robots can work without moving.

The reading process is a crucial part of humans' daily routine through which they need to get new information and stay connected with the world. However, some people are unable to read texts, such as children, people suffering from visual impairments, and blind people.

Therefore, this project is proposed in order to create a stationary robot that makes the reading process easier for the above-mentioned segments of people.

1.2 Motivation

The project is proposed in order to help people with special needs read whatever they want and keep them connected with their surroundings. Imagine how many people are suffering from the inability to read. Although there are some available solutions, such as braille books which are specially designed for the blind, these books are expensive, and, therefore, not everyone can afford them. In addition, these books may not be suitable for all groups of people who read texts [1]. Moreover, even though there are many audiobooks available on the internet, not all books people may need are available in audio [2]. As a result, there is a need to find a solution in order to make any book in the world readable for all people.

The knowledge and skills that were achieved during previous years it used to implement this project. Specifically, courses related to image processing, microcontroller, scientific research methods, as well as laboratories.

1.3 Importance

The project aims at facilitating the reading process for many disadvantaged groups of people, such as the ones suffering from visual impairments, old people, and people with reading difficulties related to age or Dyslexia. The research results show that there was a report by the World Health Organization which revealed that at least 2.2 billion people around the world suffer from blindness or low vision [3]. The project outcome will be of great significance to society as it will help people

with special needs to get the information they need easily. They will be able to listen to texts instead of reading them. In addition, the project outcome can also prove to be beneficial to children who have not learned to read yet; they can use it to listen to stories and children books about, and, therefore, motivate them to love learning and reading.

1.4 Objective

The project is expected to achieve the following objectives:

1. The robot should find the text on the page and convert the text to voice.

2. The robot turns the book pages using a wheel and an arm.

1.5 Description of the project

This project is based on the raspberry pi microcomputer, which acts as an interface between the system and the user and controls robot components like a camera, speaker, and motors.
As the camera is placed on the printer rail that is mounted on a stand, the dc motor moves the camera on the rail to fit the book pages and capture the current page then sends it to Raspberry Pi where it is analyzed and converted to voice. Then, the camera moves according to the location of the page. When the current page finishes, the Raspberry pi sends instructions to the servo motors to turn the pages, by rotating the wheel to bend the paper, and then the arm turns the page. The following photos show the page-turning mechanism:

Figure 1.5.1: The first force (F1)

Figure 1.5.2: The first force (F2)

To turn the page, two forces must influence the page of the book. The first force (F1), as shown in the figure 1.5.1, is a fraction force that is produced by sliding a page over the next one to give it a vertical elevation. The second force (F2), as shown in figure 1.5.2, pushes the lifted page to the other side of the book. Those two forces are produced using a wheel for F1 and arm for F2. [4]

1.6 Problem analysis

Some people such as the blind, old people, children, and people suffering from visual impairment have problems that prevent them from reading books like normal people. Braille books that help blind people to read are expensive and not easy to understand and learn; the Braille language can be complicated [1]. In addition to that, many books are not available in audio or digitized due to copyright issues [5].

In this project, a robot was built to read a book and turn the pages, which will facilitate the reading process for the above-mentioned groups of people.

1.7 List of requirements

The robot should:

1. Use a camera to take a capture of the current page.
2. Speak out the content that was converted into text format using the speaker.
3. Be able to turn the book pages in both directions.
4. Turn the current page to the next after completing the previous one.
5. Be easy to use by children, blind people, and people suffering from visual impairments.

1.8 Expected results

The robot is expected to:-

1. Convert text into voice and turn the pages in both directions.

2. Be comfortable and easy to use by all target groups.

Chapter 2

Background

2.1 Overview

This chapter is an introduction and a description of the hardware and software components and the system design that we will use.

2.2 Theoretical background

This section provides details about some technologies that will be used in this project.

.

2.3 Technologies to be used in the project

1. **Raspbian operating system:**
   "Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make the Raspberry Pi run. However, Raspbian provides more than a pure OS." [6].

2. **Python programming language:**
   Python is a high-level programming language that is simple to write and read and easy to learn. It also uses an object-oriented approach, python advance as an open-source. This language was used because of the libraries available for pythons such as OpenCV for image processing and RPi library that gives the access to raspberry pi General Purpose Input/Output (GPIO) and many other libraries that were used to build that application like NumPy, sys, image, pi camera, and time.

3. **OpenCV library:**
   OpenCV is one of the common libraries for computer vision. It was used to avoid low accuracy and preprocess the image for OCR. [7]

4. **Tesseract OCR**

   Tesseract is an open-source text recognition that uses Optical Character Recognition (OCR) Engine, and is the main element in our project. “It is a technology that can convert the scanning image into editable text. OCR is implemented in this project to recognize characters which are then read out by the systems through a speaker”. [8]

5. **pyttsX3 library**

   “Is a text-to-speech conversion library in Python. Unlike alternative libraries, it works offline, and it is a very easy to use tool which converts the entered text into speech”. [9]

2.4 Literature review

This section reviews some of the similar projects that were conducted to help solve the reading problems that people with reading difficulties. After extensive research for related studies and sources, we came across a limited range of solutions that are related to the proposed project.

1. **Raspberry pi based reader for blind people:**

   Books are available in abundance. Multiple projects were conducted based on text-to-speech conversion, but most of these projects did not include a page-turning element. For example, a paper done by the University of Pune in India presented an automatic document reader, developed on Raspberry Pi, for visually impaired people. This automatic document reader has a capture button that captures the document image using the OCR technology and the Text-to-Speech library, and then the text converted into an audio output can be heard by connecting headsets via 3.5mm. [10]

2. **A switch activated page turning device for people with disabilities to increase independent interaction with printed materials:**

   The University of Texas at Austin designed page-turners for people who have limited or no finger, hand, or arm dexterity, often find it difficult to turn pages and interact with the world of knowledge

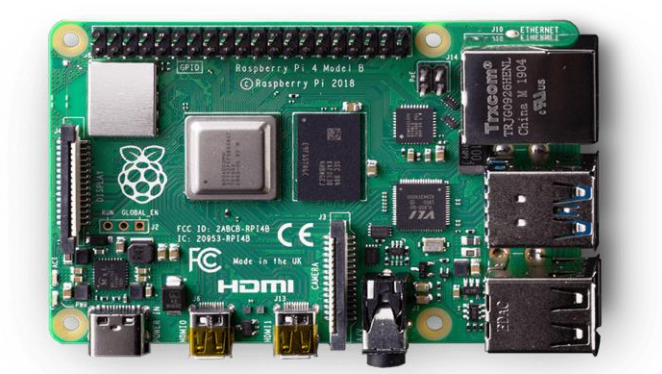
   contained in printed materials. The device performs three basic functions: support a book, hold the pages open, and turn one page at a time by using rubber friction wheel roller assembly that rolls the top page into an arc loop near the binding. And then a cam hook is used to turn to a position where the hook protrusion enters the recess under the arc and pulls the paper from underneath the wheel. [11]

This project combines the previous two ideas so; it will capture the images and convert it to audio and add an element that can turn pages to provide a book reader and page-turning robot.

2.5 Hardware components of the system

This section describes the main hardware components needed to construct this project and illustrates the function of each component and how would they integrate.

1. **Raspberry Pi 4, model B:**



The Raspberry PI "is a microcomputer that plugs into a computer monitor and uses a standard keyboard and mouse" [12]. This model was used in this project because of its capability to manipulate images in high performance. It can run Linux OS, and it can be programmed by high-level languages such as python.

Specifications of Raspberry PI [13]:

- CPU clock: 1.5 GHz
- GPU: VideoCore VI
- RAM: 2 GB DDR2
- Analog Audio Out: 3.5mm jack
- Core Type: Cortex-A72(ARM V8 64-bit)
- Pins:40-pin extended GPIO
- Micro SD port: for loading your operating system and storing data.

2. Raspberry Pi camera module v2:

It is a component that can take pictures. This component was chosen for the project because of its capability to capture images of pages in high resolution to increase the ability to find the text in these images.

3. Speaker:

It is a device used to produce audio to be heard by the listener. It is used in this project to pronounce the converted text from the image, so the user can listen and understand the captured text.

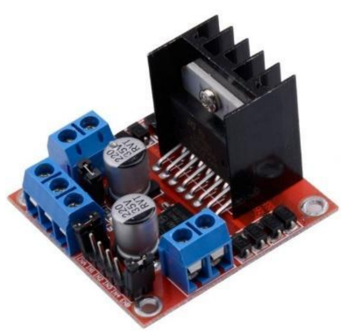
4. Servos motor:

"A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback." [14]

5. DC motor:

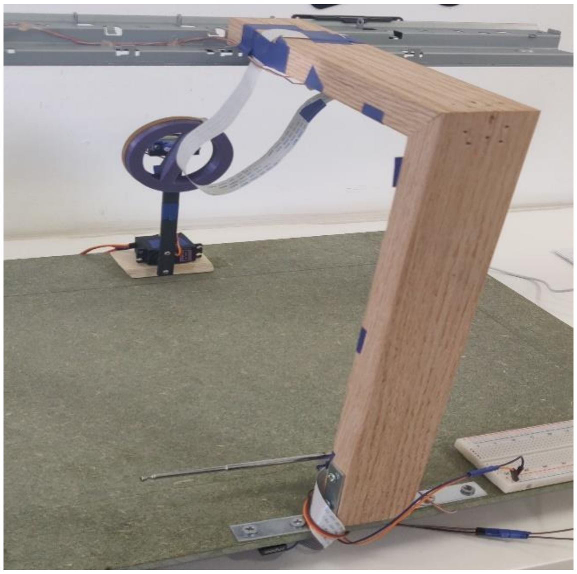
It is an electromagnetic device that converts direct current electrical energy into mechanical energy.

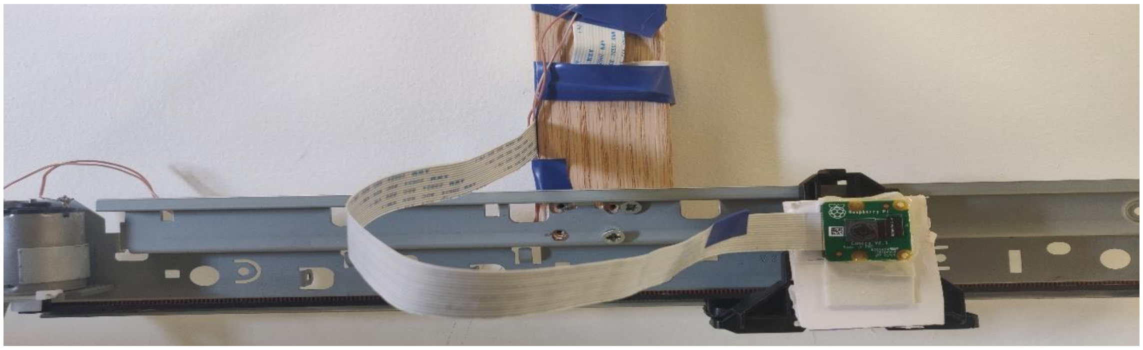
6. H-bridge module:



This typically acts as an interface between the motors and raspberry pi, and it is used to control the speed and direction of DC motors.

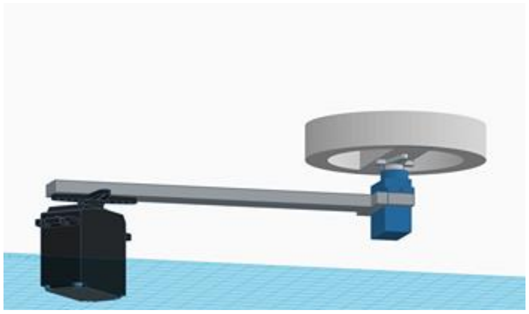
7. Camera stand and printer rail:





A wooden item that carries the printer rail which holds the camera.

8. Arms and wheel



Servo 1 and servo 2

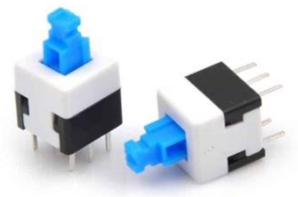
servo 3

A small printed wheel and an arm were used to bend the book pages. A light aluminum arm that is attached to servo motors was also used to turn the pages.

9. Power supply:

It is an electrical device that can supply electrical power. This model of raspberry pi requires a power supply with 2.4A and 5V.

10. Switch:



The switch will be used to turn the stationary robot on/off and used to turn the book pages in the both directions. It will be connected to the raspberry pi.

11. Connection wires:

A set of wires used to connect the components.

12. Rechargeable battery:

It is a type of electrical battery which can be charged, discharged into a load, and recharged many times.

2.6 Hardware components option

During research, many alternatives have been included, such as the following:

- **Camera options:**
  - *First option:*
    Using a USB web camera and attach it to the raspberry pi.
  - *Second option:*
    Using a raspberry pi camera module V2 and attach it to the raspberry pi.
  - *Chosen option:*

    The second option was chosen because it is designed for raspberry pi, and it less expensive. It has a capability of 3280 x 2464 pixel static images, 8-megapixel native resolution. It also provides low power mode and reduces power consumption. However, the quality of this camera is more than that of the first one.

- **H-bridge module controller options:**
  - *First option:*

    Use L298N motor drive controller board.

  - *Second option:*

    Use L293D motor drive controller board.

- *Chosen Option:*

  The first option was chosen because it weighs less, meets the requirements, needs 5V, and is widely available in the market whereas the second option is not available.

- **Microcomputer options:**
  - *First option:*

    Use raspberry Pi 4 model B microcomputer board.

  - *Second option:*

    Use raspberry pi 3b+ microcomputer board.

  - *Third option:*

    Use Arduino mega microcontroller board.

  - *Chosen option:*

    The first design option was used because it is easy to program, has better performance, manipulates images at a suitable speed, and has RAM of 2 GB DDR2 compared to the second option that has 1GB. The third has not enough amount of memory, so it cannot process images well.

- **Motors options:**
  - *First option:*

    Using Servo motors that are connected to raspberry pi.

  - *Second option:*

    Using a DC motor that is connected to the raspberry pi.

  - *Third Option:*

    Using the stepper motors that connected to raspberry pi.

  - *Chosen Option:*

    The first one was chosen because it is easily controlled with the raspberry pi and it provides a high level of torque at high speed. The second one was also chosen because it’s cheaper, and in this project, high accuracy is not required. As for the stepper motor, it has limitations at high-speeds though it loses nearly all of its torque.

- **Switch options:**
  - *First option:*
    Using push button switch.
  - *Second option:*
    Using button switch.
  - *Third option:*
    Using a touch sensor switch.
  - *Chosen option:*
    The first option was chosen. It is easy to use by blind people because it has a suitable shape and size, and it costs less than the other options.

- **Power sources option:**
  - *First option:*
    Using USB-C power supply for raspberry pi.
  - *Second option:*
    Using batteries that are connected to the raspberry pi.
  - *Chosen option:*
    The first option was chosen because it provides constant power for raspberry pi, and the second option was used to power the DC motor.

2.7 Design specifications and constraints

2.7.1 Robot rules:

It's important to give the user guide for the users, which are the following:

- The lighting on the book should be suitable.
- The robot aluminum arm will move in 180 degrees to turn the book pages, and the printed arm will move 90 degrees in the turn direction.
- The camera will change its location to fit the right and left book pages.
- The robot's motors will start running when a signal is given to start the book reading. They will start in a camera placed position on the start point of reading. Then, they will keep on reading and turn the book pages until the user turns off the switch.

2.7.2 Design specifications:

The robot must be able to move the camera to fit the position of the page, take a picture, convert the image to text using OCR, and then convert the text to voice by using the Text-to-Speech library. After it takes pictures of the book's pages on the right and the left, it sends a command to turn in the book pages using a wheel that can bend the pages and use an arm that can turn the pages.

2.7.3 Design constraints:

1. The reading process is restricted by the sizes (A4).
2. If the user does not hear any word or statement clearly and wants to rewind, they will not be able to repeat it, until repeat the current page from the start.

Chapter 3

System design

3.1 Overview

This chapter illustrates the conceptual design of the system; it also shows a block diagram of its components, flow chart of the system.

3.2 System diagram

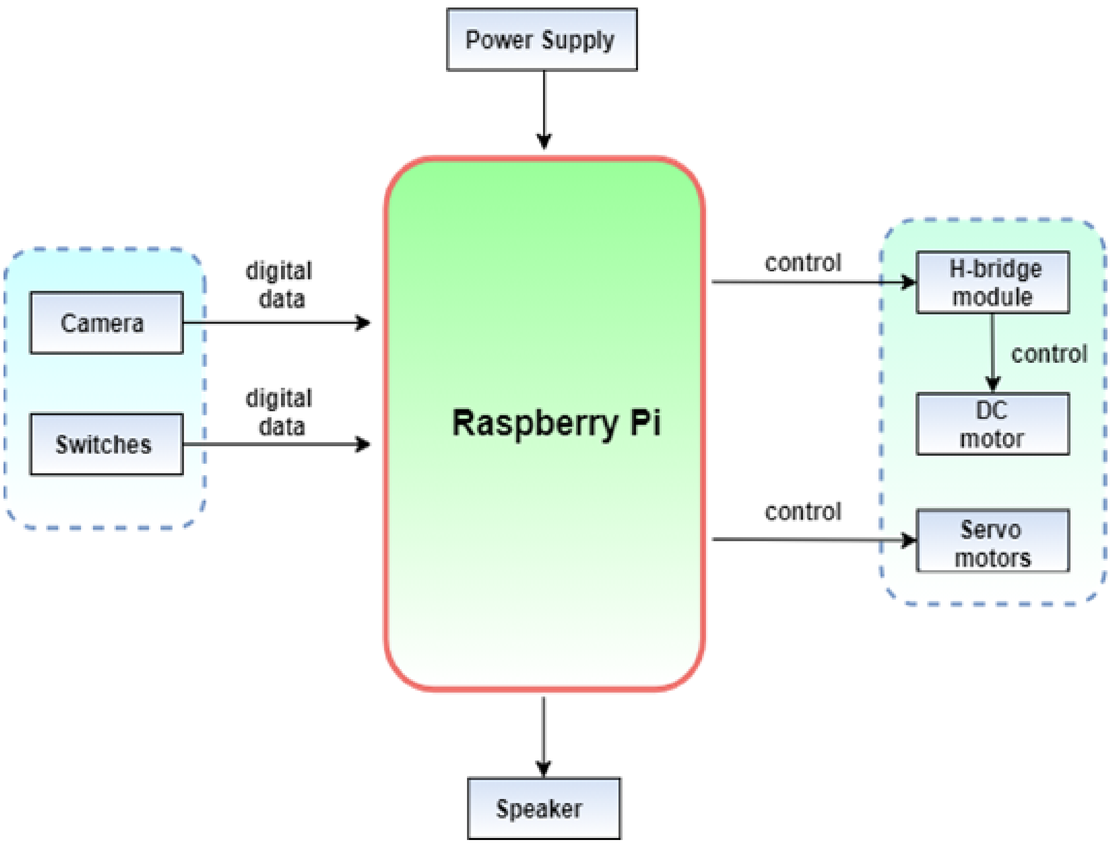


Figure 3.2: System Block Diagram

The figure shows the block diagram of the system and all the components connected to it, which it is shown the camera, switches, servo motors, and H-bridge module that control of DC motor.

3.3 Schematic diagrams:

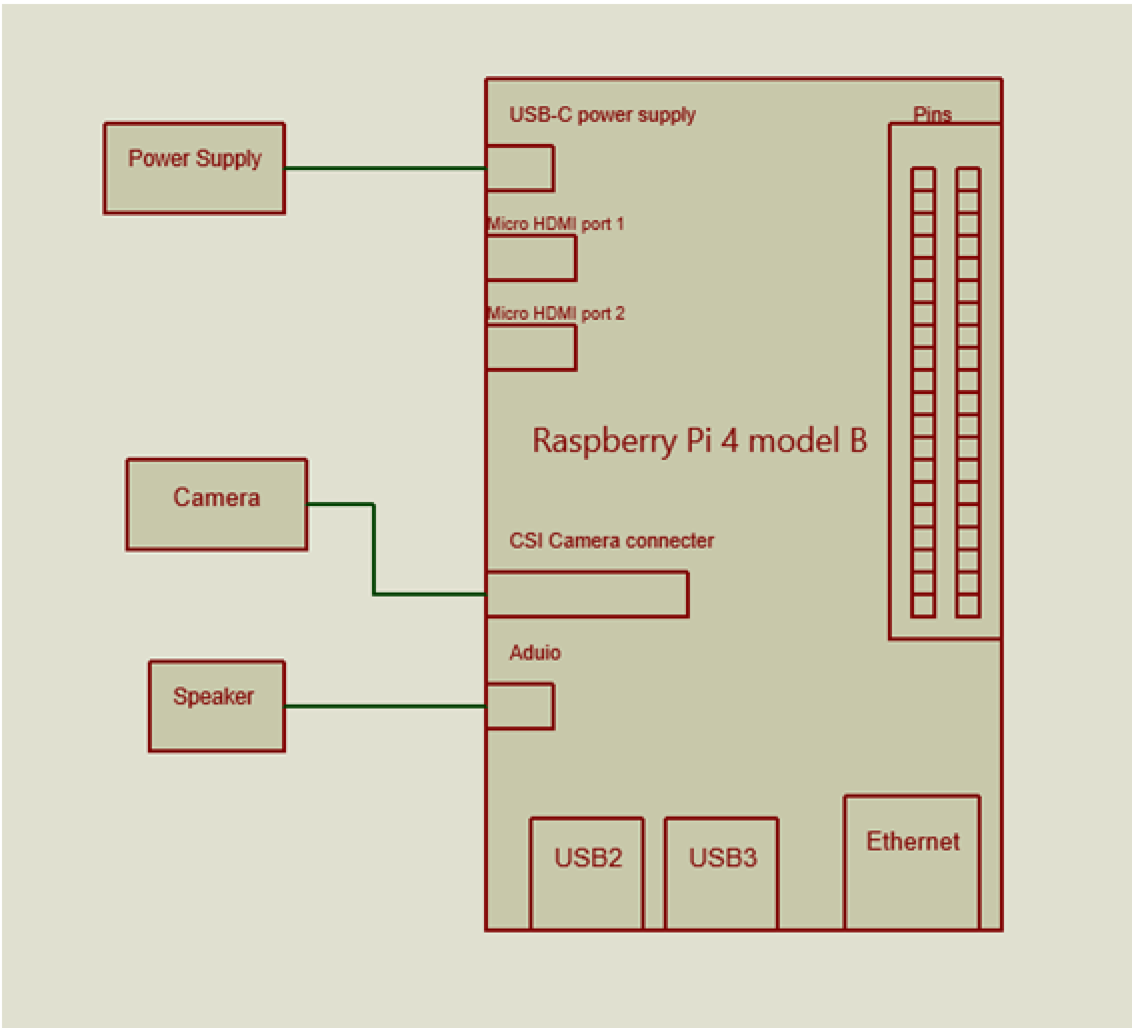


Figure 3.3.1: Schematic diagram of some parts (power supply, camera, and speaker).

The figure shows that the power supply, raspberry pi camera, and the speaker were connected to GPIO pins on raspberry pi, which acts as an interface between the user and the system.

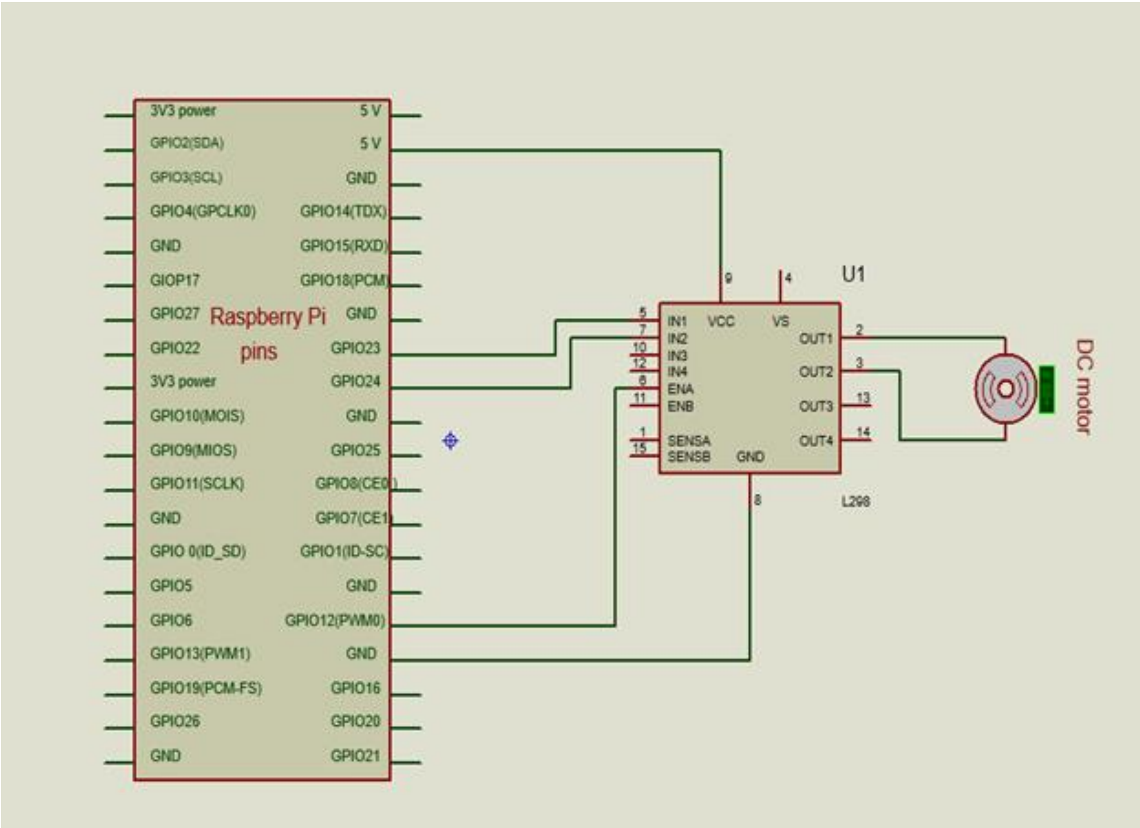


Figure 3.3.2: Schematic diagram of DC motor and motor driver.

The figure shows the H-bridge which is connected to raspberry pi GPIO pins and enabled in H-bridge (ENA) connected to PWM0 pin (stands for pulse width modulation) that will control the DC motor.

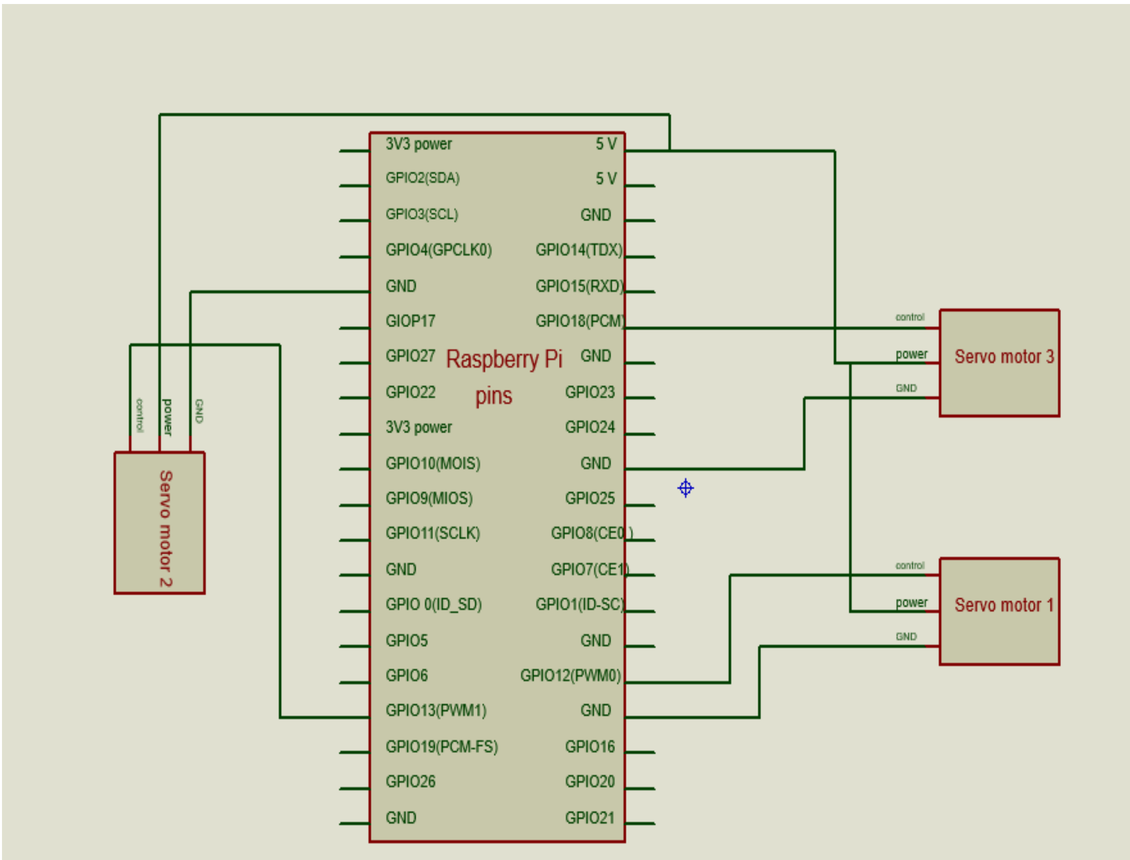


Figure 3.3.3: Schematic diagram of Servo motors.

This figure shows that the servo motors is connected to raspberry pi GPIO pins in which the control pins in servo motors are connected to (PWM) pins.

3.4 System flow chart

The flow chart shows that this project can turn the book pages in both directions. The direction of the turning can be activated by pressing the appropriate switch, either right or left. After the camera moves to the suitable place as shown in the figure, the process of turning pages will be accomplished by moving each motor in the appropriate direction, then the reading process will activate, by taking a capture of the page, after that it will convert the image to text using OCR and the pyttsx3 library will be used to convert text to speech through the speaker. The system will repeat the previous process until it is turned off.

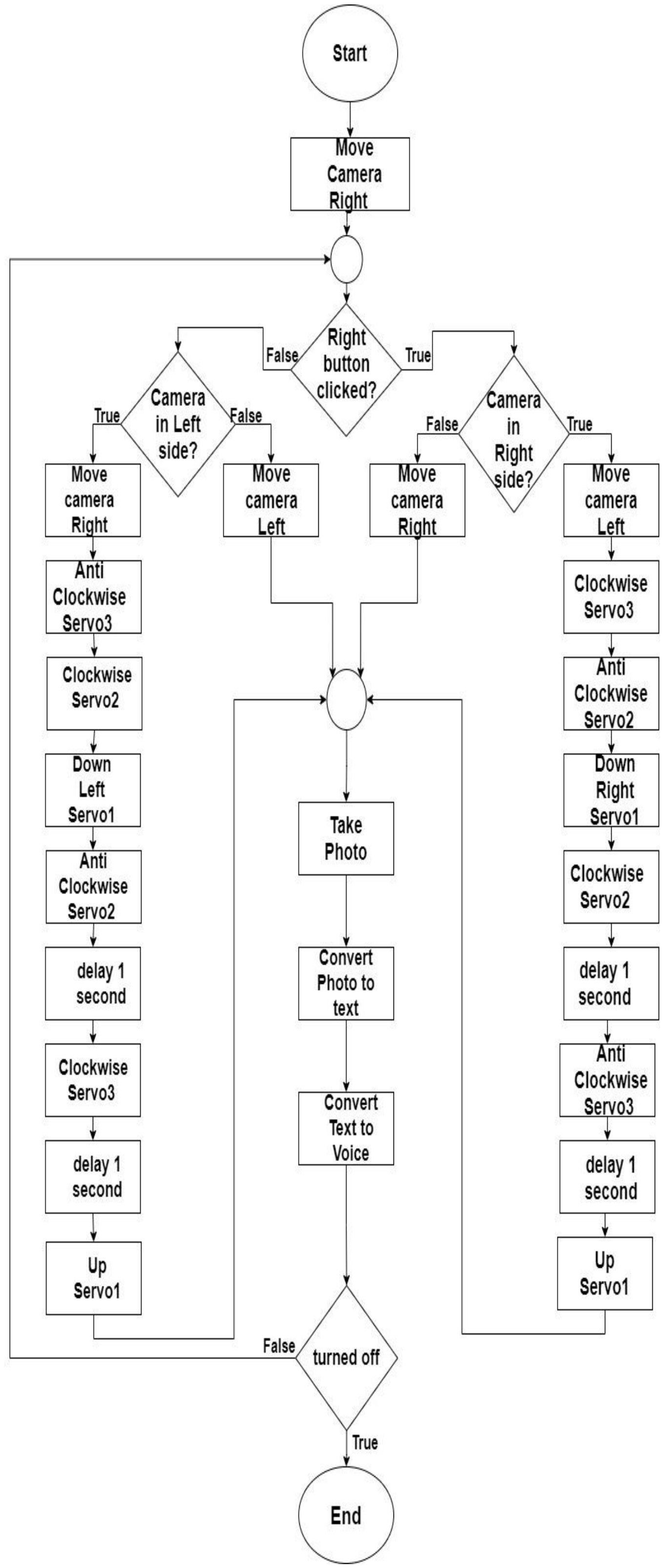


Figure 3.4: The Flow chart of the system

Chapter 4

Software and hardware implementation

4.1 Overview

This chapter describes the project software and hardware implementation as well the different components and tools used to build the robot.

4.2 Raspberry pi microcomputer

To download the operating system on raspberry pi, a NOOBS zip file was used. NOOBS (New Out Of the Box Software) is a simple operating system install manager for the raspberry pi and other operating systems; so, a 32GB class10 SD card was used and formatted before copying the NOOBS file. Then, the latest NOOBS 3.5.0 file was installed and put it on the computer.

4.3 Installing needed packages

4.3.1 OCR and OpenCV implementation

Pytesseract is a wrapper for Tesseract-OCR Engine that can convert various types of images to text by using the python library tesseract. The latest version of the tesseract (Pytesseract 0.3.6) was installed, which is a python tool that is based on deep learning so its accuracy is very high [15]. However, because of some camera issues, the image-to-text conversion is not accurate; so, the pre-process was required, for that OpenCV was installed. At first, the correct Python virtual environment was activated. Then, the Python shell was entered and the OpenCV was tested.

4.3.2 Text to speech implementation

To convert text to speech, pyttx3.2.9 version library was installed and generating the digital audio It supports espeak engine which runs on every other platform like Raspberry Pi OS.

To implement text-to-speech with Python script by writing lines of code, various functions were used:

1. runAndWait () function: this function keeps track of when the engine starts converting text to speech and waits for that much time, and does not allow the engine to close. [16]

2. getProperty () function: this function changes the voice of the pyttsx3 engine, and it get the list of objects of voices that takes a string as a parameter and returns an object matching the string. [16]

3. setProperty() function: this function takes the property name and the changing property id. [16]

4. say () function: this function takes a string value and speaks it out. [16]

4.4 Hardware implementation

This section will provide some information about the hardware implementations done through our project:

4.4.1 Power configuration

The raspberry pi needs power to run all of its elements, so we used a 15.3W USB-C power supply.
The DC motor driving the rail needs an external power supply therefor, a 12v lead acid battery was used.

4.4.2 Raspberry pi camera configuration

The camera was needed to take a capture of the book pages to convert it to voice. The raspberry pi camera is connected to the camera module port of the raspberry pi.

The pi camera module has a focal length of about 50cm, but the distance between the camera and the book is 30cm, so the focus of the camera needs to be changed by rotating the lens to reduce the focal length of the camera, and the maximum resolution is 2592×1944 for still photos.

4.4.3 Printer rail implementation

For more accuracy, a printer rail was used and the camera was placed on it to move the camera left and right to fit the pages of the book. By using the DC motor that is controlled by the motor driver (H bridge).

The motor driver (H-bridge) is connected to GPIO pi pins to control the direction of rotation of the motor, and use PWM to control the speed of the motor, and for this, a python script was written that toggle GPIO pins HIGH or LOW.

4.4.4 Turning page process implementation

For the whole process, three servo motors were used. Every servo motor connects with a single GPIO. By writing a python script that sets a reference angle for every servo.

To bend the book pages, it needs a fraction force, which was applied by used two servo motors. The first servo MG996R holds the second servo by using a printed arm and press the paper to give more fraction force. If the right switch is pressed, the servo moves to the 0°, and then back to 90°. If the left switch is pressed, it moves from 90° to 180° to left.

The second micro servo 9G had a wheel placed on it, and the second was 3D printed with rubber wrapped to it. Then, moves anticlockwise from 180° to 0°.

The third servo is a micro servo 9G that holds the light aluminum arm, and it is responsible for turning the paper by pushing it to the turning page side. The servo is controlled by sending PWM signals to determine the servo position.

4.5 System pseudo code

Function Move Camera to the Left side

Pass In: nothing

Turn the DC motor to move camera to the left side.

Pass Out: nothing

Endfunction

Function Move Camera to the Right side

Pass In: nothing

Turn the DC motor to the right side and turn it off after five seconds.

Pass Out: nothing

Endfunction

Function clockWise Servo number_3

Pass In: nothing

turn the aluminum servo clockwise
by set angle to zero degree.

Pass Out: nothing

Endfunction

Function anti clockWise Servo number_3

Pass In: nothing

turn the aluminum servo anti clockwise
by set angle to 180 degree.

Pass Out: nothing

Endfunction

Function clockWise Servo2

Pass In: nothing

turn the wheel servo clockwise
by set angle to 0 degree.

Pass Out: nothing

Endfunction

```
Function Anti clockWise Servo2

   Pass In: nothing

     turn the wheel servo anti clockwise
      by set angle to 180 degree.

   Pass Out: nothing

Endfunction

Function Down Right Servo1

   Pass In: nothing

     turn the servo that hold wheel to the right
      by set angle to 0 degree.

   Pass Out: nothing

Endfunction

Function Move up Servo1

   Pass In: nothing

     turn the servo that hold wheel to the normal position
      by set angle to 90 degree.

   Pass Out: nothing

Endfunction

Function Down Left Servo1

   Pass In: nothing

     turn the servo that hold wheel to the left
      by set angle to 180 degree.

   Pass Out: nothing

Endfunction
```

```
Function Take a photo

    Pass In: nothing

        take photo of page using pi camera.

    Pass Out: nothing

Endfunction

Function Convert Photo to Text

    Pass In: photo

        using pytesseract to convert image to string.

    Pass Out: text

Endfunction

Function Convert Text to Voice

    Pass In: text

        by using engine in pyttsx3 convert the text to voice
        or using gTTS (google text to speech API) and play it.

    Pass Out: nothing

Endfunction
```

```
Function main
    WHILE TRUE THEN
        IF right switch pressed THEN
            IF camera in the right side THEN
                call: Move Camera to The Left side
                call: ClockWise Servo number_3
                call: Anti clockWise Servo2
                call: Down Right Servo1
                call: ClockWise Servo2
                call: Anti ClockWise Servo number_3
                call: Move Up Servo1
            ELSE
                call: Move Camera to The Right side
            ENDIF
        ELSE
            IF Left switch pressed THEN
                call: Move Camera to The Right side
                call: Anti ClockWise Servo number_3
                call: ClockWise Servo2
                call: Down Left Servo1
                call: Anti clockWise Servo2
                call: ClockWise Servo number_3
```

```
            call: Move Up Servo1
        ELSE
            call: Move Camera to The Left side
        ENDIF
    ENDIF
    call: Take a Photo
    call: Convert Photo to Text
    call: Convert Text to Voice
  Endwhile
Endfunction
```

Chapter 5

Validation and testing

5.1 Overview

This chapter presents the results of the system implementation of this project.

5.2 Motors test and connection

Each servo motor connects separately to raspberry pi with power and then it is run the code. Import required python libraries like RPi GPIO and time, the GPIO type chosen in servo motor is PWM. After the code ran more than once and monitored the motor movement, the suitable reference angle was detected.

For dc motors was tested by connected to the battery, and the H-bridge was connected it to both dc and raspberry pi. Then, a python script was written to be run.

5.3 Camera test and connection

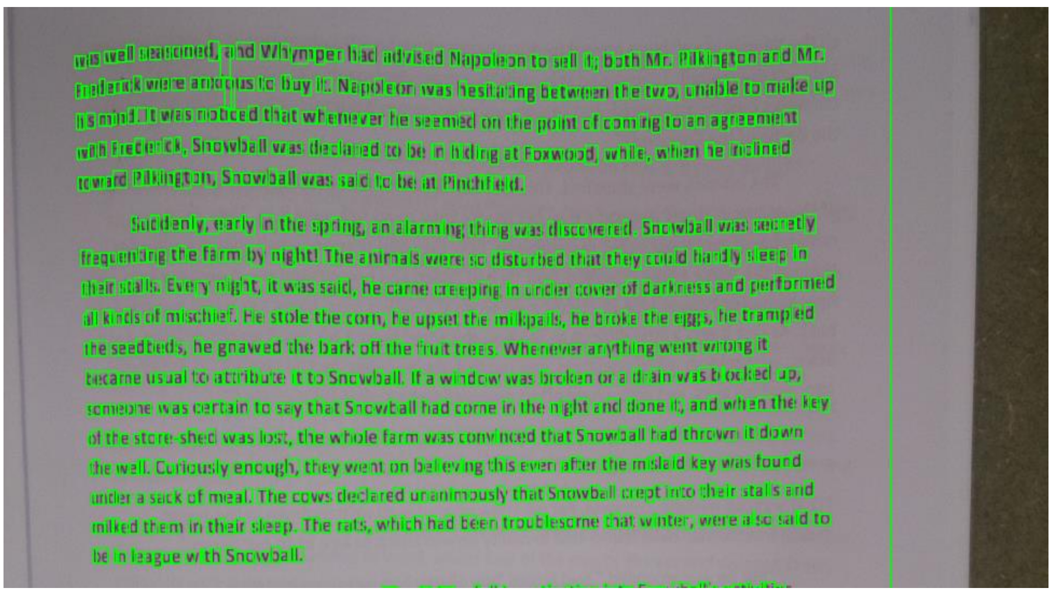
To test if the Pi camera are working correctly, the first step is connecting it to the camera module port on Raspberry Pi using a camera module ribbon cable that fits the pi camera module port.
Then the camera was enabled by Raspberry pi configuration and in the terminal window, the `raspistill -o Desktop/image.jpg` command was written to take the image and save it to the Desktop in a jpg format.

5.4 OpenCV test

OpenCV library that will be used to provide pre-processing to the image. We install OpenCV library by following the instructions in [17]
After that, it was tested trying the below command in the python terminal

```
$ cd ~
$ workon cv
$ python
>>> import cv2
>>> cv2.__version__
'4.1.1'
```

Then the text was detected by putting a boundary box around each litter in the image to ensure that each letter will be recognized by OpenCV and Tesseract.


was well seasoned, and Whymper had advised Napoleon to sell it; both Mr. Pilkington and Mr. Frederick were anxious to buy it. Napoleon was hesitating between the two, unable to make up his mind. It was noticed that whenever he seemed on the point of coming to an agreement with Frederick, Snowball was declared to be in hiding at Foxwood, while, when he inclined toward Pilkington, Snowball was said to be at Pinchfield.

Suddenly, early in the spring, an alarming thing was discovered. Snowball was secretly frequenting the farm by night! The animals were so disturbed that they could hardly sleep in their stalls. Every night, it was said, he came creeping in under cover of darkness and performed all kinds of mischief. He stole the corn, he upset the milkpails, he broke the eggs, he trampled the seedbeds, he gnawed the bark off the fruit trees. Whenever anything went wrong it became usual to attribute it to Snowball. If a window was broken or a drain was blocked up, someone was certain to say that Snowball had come in the night and done it, and when the key of the store-shed was lost, the whole farm was convinced that Snowball had thrown it down the well. Curiously enough, they went on believing this even after the mislaid key was found under a sack of meal. The cows declared unanimously that Snowball crept into their stalls and milked them in their sleep. The rats, which had been troublesome that winter, were also said to be in league with Snowball.

5.5 Speaker and sound output test on raspberry pi

To make raspberry pi speak, at first, the audio was configurated in raspberry pi. Then, the espeak was installed, the mp3 audio was on and the sound was out.

```
sudo apt-get install espeak
```

Following that, the pyttsx3 was installed on raspberry pi to try this code:

```
import pyttsx3

engine = pyttsx3.init()

engine.say("I will speak this text")

engine.runAndWait()
```

5.6 Implementation issues

5.6.1 Hardware issues

1. There are different turning pages techniques, and we meet a lot of defaults to choose one of them, at first, a stepper motor was used to bend the book pages, but its precision and its max speed were not fast enough to bend the paper. Therefore, it was replaced with two servo motors: the first one holds the wheel and the second one holds the connecting arm with the motor. Consequently, the robot now has a tool with compressive strength sufficient for bending the book pages.

2. Another issue was determining the appropriate connections to use to connect the raspberry pi to the screen. Raspberry pi needs a micro–HDMI TYPE D to HDMI female converter adapter and a HDMI to DVI adapter. When connecting Raspberry pi to a monitor, the monitor started to blink. After research, we found that HDMI to DVI needs high power while the raspberry pi needs power low, so the HDMI to DVI was replaced by a VGA HDMI Gold Male to VGA HD-15 Male Cable HDMI-VGA.

3. To solve camera accuracy issues that caused an error rate in the results of the OCR, a single page was captured instead of two by putting a camera on a printer rail. In addition, the camera captures were reversed crosswise, then the image was processed by rotating it using a function from OpenCV.

4. One of the most important aspects of the project is image processing, therefore, the issue of rapid increase in the raspberry pi process's temperature should be addresses to avoid the probability of its failure. In that respect, an aluminum cooler and a few fans were added to reduce the raspberry pi process' temperature.

5.6.2 Software Issues

1. When installing OpenCV, in the $ make -j4 step, it stopped downloading the OpenCV library. After searching for a solution to this problem, it was found that the ram class has a big impact on the process; so class 4 ram was used and replaced it with class 10. That way, the problem was solved and download was completed.

2. Since the project works without the internet, pyttx3 was used. The only way to change and control voices is to change the property because of the initial results were not good. Therefore, an internet option was added to enable using gTTS for better voice control.

5.7 System validation

The system was able to turn in both direction, and it turn one page at a time, but sometimes it turns more than one page because of the fraction force that changes by changing the book height. For the image-to-text conversion, for every 700 words, approximately 20 words are missed. The system was able to convert all the text into audio, but the accuracy of the voice was not high.

Chapter 6

Conclusion and future work

6.1 Overview

This chapter shows the project conclusion, the main achieved objectives, and will propose future works that can be developed on the system.

6.2 Conclusion

In conclusion, the two main objectives of this project were achieved. The robot can find texts in the book and convert them into sound it also can turn the pages of the book using a wheel and arm that are controlled by the motors. This was achieved by using the appropriate hardware components, as well as installing the required packages. The project's hardware and software have been successfully integrated and work to meet the requirements.

6.3 Future Work

Some future works are suggested and recommended to improve the Book Reader and page-turner robot:

- Develop the robot by making it read all kinds of books that differ in size, thickness, and type of font.
- Using a high-resolution camera to improve the reading process.
- Using artificial intelligence to develop the project like make it read the stories with emotion.

References

[1] Bit, A., n.d. Design and Development of Affordable Healthcare Technologies. In the United States of America: IGI Global, p.233.

[2] National Aphasia Association. 2017. How To Turn E-Books Into Audio Books - National Aphasia Association. [Online] Available at:

https://www.aphasia.org/stories/turn-e-books-audio-books/. [Accessed 6 May 2020].

[3] Who.int. 2019. WHO Launches First World Report On Vision. [Online] Available at:

https://www.who.int/news-room/detail/08-10-2019-who-launches-first-world-report-on-vision. [Accessed 6 May 2020].

[4] S., 2018. DEVELOBMENT OF VOICE CONTROLLED AUTOMATIC PAGE

TURNING SYSTEM. International Journal of Research in Engineering and Technology, 07(06), pp.86-87.

[5] Onlinebooks.library.upenn.edu. 2020. How To Put Books Online. [Online] Available at:

https://onlinebooks.library.upenn.edu/bookinstr.html. [Accessed 6 May 2020].

[6] (2021) Raspberrypi.org. Available at:

https://www.raspberrypi.org/software/. [Accessed: 10 January 2021].

[7] About - OpenCV (2021). Available at:

https://opencv.org/about/. [Accessed: 10 January 2021].

[8] Phutak, V., Kamble, R., Gore, S., Alave, M., & Kulkarni, R. R, 2019. Text to Speech

Conversion using Raspberry-PI.International Journal of Innovative Science and Research Technology, pp.291.

[9] PyPI. 2020. Pyttsx3. [Online] Available at: https://pypi.org/project/pyttsx3/. [Accessed 11 May 2020].

[10] Jadhav, M., Koli, S. and Kulkarni, S., 2018. Raspberry Pi Based Reader for Blind. International Journal of Innovative and Emerging Research in Engineering.

[11] Bailey, M., Banks, J. and Pattarkine, V., 2003. A SWITCH ACTIVATED PAGE TURNING DEVICE FOR PEOPLE WITH DISABILITIES TO INCREASE INDEPENDENT INTERACTION WITH PRINTED MATERIALS. [Online] Resna.org. Available at: https://www.resna.org/sites/default/files/legacy/conference/proceedings/2003/Papers/SDwinners/Bailey2_TSP.htm . [Accessed 11 May 2020].

[12] Raspberry Pi. 2020. What Is A Raspberry Pi? . [Online] Available at: https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/ [Accessed 28 May 2020].

[13] Raspberry PI models comparison. 2020. [Online] Available at: https://socialcompare.com/en/comparison/raspberrypi-models-comparison. [Accessed 11 May 2020].

[14] Servomotor (2016). Available at: https://en.wikipedia.org/wiki/Servomotor. [Accessed: 10 January 2021].

[15] [Tutorial] OCR in Python with Tesseract, OpenCV and Pytesseract (2020). Available at: https://nanonets.com/blog/ocr-with-tesseract/. [Accessed: 10 January 2021].

[16] Using pyttsx3 — pyttsx3 2.6 documentation (2021). Available at: https://pyttsx3.readthedocs.io/en/latest/engine.html [Accessed: 10 January 2021].

[17] Rosebrock, A. (2019) Install OpenCV 4 on Raspberry Pi 4 and Raspbian Buster - PyImageSearch, PyImageSearch. Available at: https://www.pyimagesearch.com/2019/09/16/install-opencv-4-on-raspberry-pi-4-and-raspbian-buster/. [Accessed: 10 January 2021].