Chapter 1

Introduction

1.1 Introduction

Artificial Intelligence is the simulation of human intelligence processes by machines. Artificial Intelligence analyzes the data & makes predictions of output. Artificial intelligence makes it possible for machines to learn from experience, adjust to new inputs and perform human-like tasks. AI has created some wide range applications in a variety of domains.

Particular applications of AI include speech recognition and machine vision. AI works by combining large amounts of data with fast iterative processing and intelligent algorithms. Allowing the software to learn automatically from patterns or features in the data. Python is widely used for artificial intelligence with packages for a number of applications including General AI, Machine Learning, Natural Language Processing and Neural Networks.

1.2 Relevance of the Project

The number of visually impaired people is growing over the past decades. As reported by the world health organization (WHO), about 285 million people worldwide are estimated to be visually impaired. To help them with their daily activities, Smart Cap is designed to narrate the description of a scene via voice assistance.

Blind people can’t experience the world the way we do. Smart Cap aims to provide this missing experience for them. Visually impaired people can’t recognize the object without touch. Smart cap aims to bring the beautiful world as a narrative to the visually impaired. Smart Cap uses webcam to generate the frames. Smart Cap contains Raspberry Pi which sends these frames to the server. Server processes these frames and classifies the objects detected in image and tags them with appropriate labels. The narrative is generated by converting the scenes in front of them to text. The experience is powered by the voice assistance. Raspberry Pi will Speake the text as an output to the user via a headset.

For the first prototype of the system, one line along with some keywords would be played as audio to the user, but in later versions a detailed description would be added as a feature.

1.3 Problem Statement: -

It is very difficult for blind people to navigate the outdoor world that they’re unaccustomed to. It is necessary to help them in navigating the surroundings. So, there is need to provide an assistant for visually impaired which narrates the description of scene.

Chapter 2

System Study

2.1 Existing System Study

1. NonVisual Desktop Access (NVDA) software

Referred Paper:

NVDA (Non-Visual Desktop Access) is an open source screen reader for Microsoft Windows which enables blind and vision impaired people to use computers. A screen reader is basically a layer on the top of the operating system which intercepts every input and output and presents it to the user in a useful way either in audio or braille format. It provides support for 12 Indian languages. NVDA can be installed directly on to your computer, but it has the ability to run entirely from a USB stick or other portable media. Braille is a tactile writing system used by people who are visually impaired.

NVDA fetches information about GUI widgets, textual data, events, user interactions and OS events with the help of Accessibility APIs, Native APIs, Application Specific APIs & OS Functions.

2. Orcam Smart Glass

OrCam devices such as *OrCam MyEye* are portable, artificial vision devices that allow visually impaired people to understand text and identify objects through audio feedback, describing what they are unable to see. The OrCam MyEye is a discreet smart camera that sits on any pair of glasses frames and connects to a portable computer designed to fit in the wearer's pocket. It's a standalone device that consists of a small smart camera and a base unit connected by a thin wire.

The OrCam device has the ability to read printed text in real time from any surface. A person can read books, newspapers, signs, labels, menus and even text on a smartphone or computer screen. It is hard for people who are visually impaired to get information. OrCam is the solution as the camera is able to read the text it "sees" and relay the information to the user. “In other words, it’s a camera that looks and talks.

3. Smart Blind Stick

The smart blind stick automatically detects the obstacle in front of the person and give him a response to the person by vibrating the stick and also with a warning sound. It detects and sounds a different buzzer if it detects water and alerts the blind. One more feature is that it allows the blind to detect if there is light or darkness in the room. The system has one more advanced feature integrated to help the blind find their stick if they forget where they kept it.

2.2 Objective: -

- To provide a complete system which will help blind people in navigating the surroundings.
- To enable blind people to perform many job functions.
- To overcome the overhead of remembering the positions of surrounding things.
- To provide a personal assistance to the blind person.

2.3 Proposed System: -

The Smart Cap consists of various modules which are deployed on the cap(client) and some of them are included in a server which allows blind person to know the things in surrounding. And the assistance is provided using voice.

- **Client**
    - ➢ Generates the frame and sends it to the server for further processing.
    - ➢ It includes Creative USB Webcam, Raspberry Pi 3 and Headset.
    - ➢ It uses free python library to generate speech for given text.
- **Server**
    - ➢ Runs object detection algorithm on received frames from client.
    - ➢ Assigns labels to the detected objects and stores these labels in dynamic list.
    - ➢ Converts all the in form of text and sends this text to the client.

- **Advantages of proposed System**

  The new system must provide the following features

  - It provides accurate narration of the scene.
  - It allows user to perform many job functions.
  - It assists user using voice assistance.

Chapter 3

Information gathering and analysis

3.1 Literature Survey

Previous Work on Existing Systems for Blind People: -

- **Paper Name:** Smart Glasses for the Visually Impaired People
- **Author Name:** N. Sriskanthan, F. Tan, A. Karande
- **Theme:** This paper presents a new design of assistive smart glasses for visually impaired students. The objective is to assist in multiple daily tasks using the advantage of wearable design format. As a proof of concept, this paper only presents one example application, i.e. text recognition technology that can help reading from hardcopy materials. The building cost is kept low by using single board computer raspberry pi 2 as the heart of processing and the raspberry pi 2 camera for image capturing. Experiment results demonstrate that the prototype is working as intended

- **Paper Name:** Computer Access for Persons Who Are Blind or Visually Impaired.
- **Author Name:** I Environmental Research Institute of Michigan, Arlington, Virginia
- **Theme:** Suitably adapted computers hold considerable potential for integrating people who are blind or visually impaired into the mainstream. The principal problems that preclude the achievement of this potential are human factors issues. These issues are discussed, and the problems presented by icon-based interfaces are reviewed. An argument is offered that these issues, which ostensibly pertain to the blind or visually impaired user, are fundamental issues confronting all users. There is reason to hope that the benefits of research into the human factors issues of people with vision impairments will also extend to the sighted user.

- **Paper Name:** Effective Fast Response Smart Stick for Blind People.
- **Author Name:** Ayat Nada, Samia Mashelly, Mahmoud A. Fakhr, and Ahmed F. Seddik.
- **Theme:** The smart stick comes as a proposed solution to enable them to identify the world around. In this paper we propose a solution, represented in a smart stick with infrared sensor to detect stair-cases and pair of ultrasonic sensors to detect any other obstacles in front of the user, within a range of four meters. Moreover, another sensor is placed at the bottom of the stick for the sake of avoiding puddles. Speech warning messages and the vibration motor are activated when any obstacle is detected.

3.2 Referred Journal Papers

Paper 1:

- **Paper Name -** Challenges faced by students with blindness.
- **Authors**- Cuthbert Majoni and Julieth Mashatise
- **Theme**- In this paper, author introduced Challenges faced by students with blindness. Blind students experience challenges that impact on their academic success when studying through ODL. Educational institutions and families had been found wanting in terms of support both financially and morally. Blind students without adequate support are unlikely to succeed in their studies. This research sought to find out the challenges faced by a student with acquired blindness enrolled at Mashonaland Central Regional Campus.

Chapter 4

Project Scope Statement and Analysis

4.1 Project Scope Statement:

Scope Description:	**In Scope:** 1. Smart Cap can be used by all blind people while navigating. 2. Can be used to improve ability of blind person to perform many job functions. **Out of Scope:** 1. Smart Cap will not work properly in deep dark place. 2. Smart Cap won't be much useful for normal people.
Project Deliverable:	Smart Cap
Acceptance Criteria:	Smart Cap should recognize objects around user.

Table 6.1. Project Scope Statement

4.2 Software Requirement Analysis and Specification

4.2.1 Resource Requirements:

4.2.1.1 Software Requirements:

1] Ubuntu 18.04 LTS: -

- Ubuntu is a complete Linux operating system freely available with both community and professional support.
- LTS is a Long-Term Support. The LTS designation applies only to specific subsets of the Ubuntu archive. The LTS may not apply to all flavors and remixes of Ubuntu.

2] Raspbian Buster: -

- 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 your Raspberry Pi run.

3] TensorFlow library (v.2.0 GPU): -

- TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks.
- It is a symbolic math library, and is also used for machine learning applications such as neural networks.

4] Python3: -

- Python is an easy to learn, powerful programming language.
- It has efficient high-level data structures and a simple but effective approach to object-oriented programming.

5] nvidia CUDA (v.10.0): -

- CUDA is a parallel computing platform and programming model developed by Nvidia for general computing on its own GPUs (graphics processing units).
- CUDA enables developers to speed up compute-intensive applications by harnessing the power of GPUs for the parallelizable part of the computation.

6] OpenCV (v.4.1.1): -

- OpenCV is a cross-platform library using which we can develop real-time computer vision applications.
- It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.

7] pyttsx3(Python Text to Speech): -

- pyttsx3 is a text-to-speech conversion library in Python.
- It works offline and is compatible with both Python 2 and 3.

4.2.1.2 Hardware Requirements:

1] Raspberry-Pi: -

- Board: Raspberry Pi 3 Model B
- Processor: Broadcom BCM2837
- CPU Core: Quadcore ARM Cortex-A53, 64Bit
- RAM: 1GB
- Wireless Connectivity: 802.11n wireless LAN (Wi-Fi) and Bluetooth 4.1.
- Power Supply (Current Capacity): 2.5 A.

2] Headset: -

- 3.5mm Jack

3] Creative Webcam: -

- USB 2.0 Webcam with 180 degree coverage.

4] Power Bank: -

- 5v(10000mah)

4.3 Requirement Analysis

4.3.1 Functional Requirement-

1. **Recognize Objects**

- Background Information

Smart cap should recognize objects around user.

1.1 Action/Response sequence.

- Description

Actor	System
Purpose	The aim of this function is to identify all the object around the user.
Precondition	The system should be ready to generate frame.
Trigger	The user should activate the system.

2. **Store Labels in List:**

- Background Information

  Smart cap should store all the labels of objects in list.

1.2 Action/response sequence:

- Description

Actor	System
Purpose	The aim of this function is to store all the labels of objects in list
Precondition	The system should assign labels to the objects.
Trigger	The Server should store it.

3. **Play Text as an Audio:**

   - Background Information.

     Speech engine should play text of a description of scene as an audio.

1.3 Action/Response Sequence:

- Description

Actor	System
Purpose	The aim of this function is to convert text into speech.
Precondition	Speech engine should be ready to say.
Trigger	The system should play audio.

4.3.2 Non-Functional Requirement-

1. Space on cap:

We need to place some components on the cap. So, there must be enough space on cap so we can place the components.

2. Exact narration:

Smart cap should provide exact narration of the scene. So, user can easily understand the situation and travel without any accidents.

3. Smooth processing:

Smart cap should provide smooth processing environment. That means system should perform the all operations within a time without any errors.

Chapter 5

System Design and Analysis

5.1 Project Module and Architecture

Modules:

1. Client:

- **Camera Module:**
  - It is connected to a Raspberry Pi.
  - Raspberry Pi generates a frame using camera and sends it to the server for further processing.
- **Text to Speech Module:**
  - Raspberry Pi converts the text received from server into the speech.
  - This speech is played by the Sound Module.
- **Sound Module:**
  - The assistance is given by headset which is directly connected to Raspberry Pi.

2. Server:

- **Object Detection Module:**
  - The further processing is done on the frame received from Raspberry Pi.
  - Object Detection algorithm is applied on the frame.
  - If no objects will get detected then that frame will be discarded.
- **Object Labeling Module:**
  - The labels assigned to all the detected objects.
  - These labels are stored in dynamic list and converted to the text.
  - Labels text is sent to the Raspberry Pi for converting it into voice.

System Architecture:

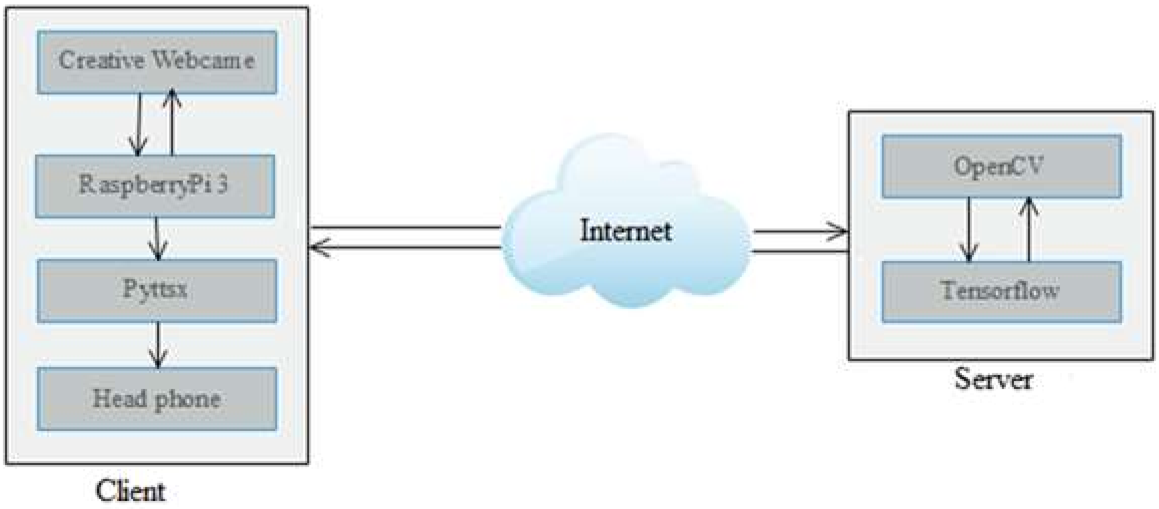


Fig 5.1.1. Architecture of Smart Cap: Vision for The Visually Impaired

5.2 System Design and Modeling

5.2.1 Static modeling:

- **Class diagram**

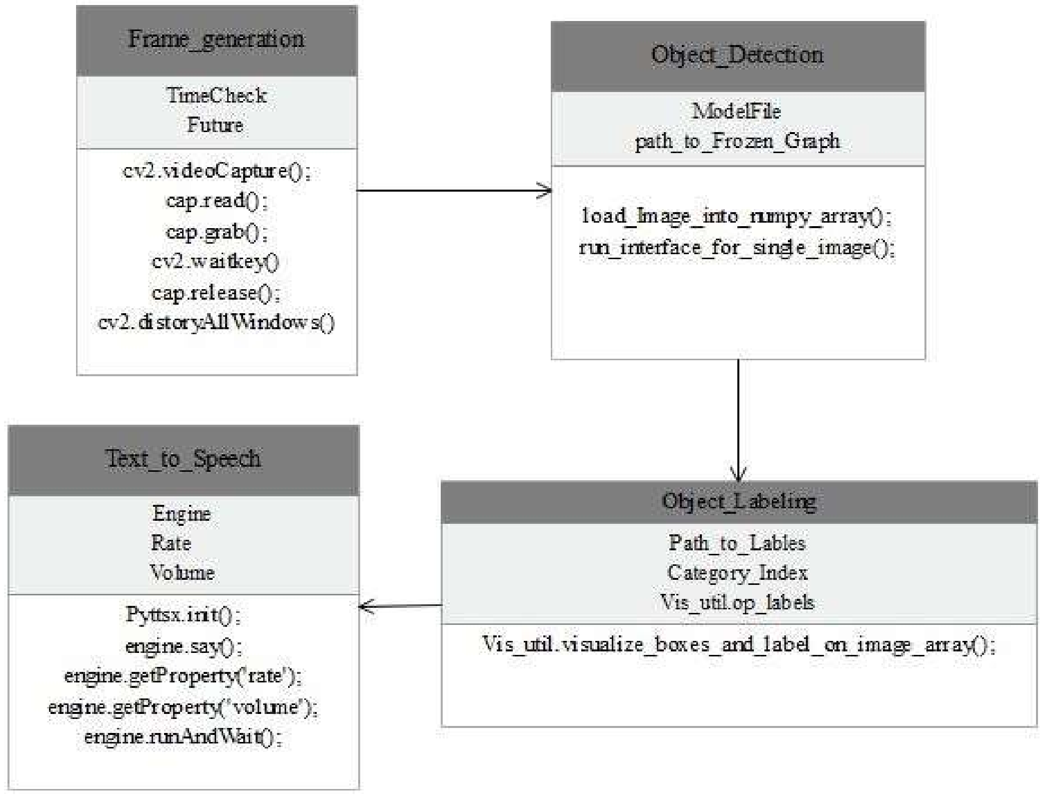


Fig 8.2.1.1 Class Diagram for Smart Cap

- In class diagram there are several Python classes used for processing of image and to give output as speech. There are some main classes like Frame_generation, Object_Detection, Object_Labeling and Text_to_Speech.
- Frame_Generation class consists of variables TimeCheck, Future and functions like cv2.videocapture(), cap.read(), cap.grab(), cv2.waitkey(), cap.release(), cv2.distroyAllWindows() which controls the generating of frames with respect to time. Object_Detection class consists of variables ModelFile, Path_to_Frozen_Graph and functions like load_Image_into_numpy_array(), run_interface_for_single_image() which detects the specific objects within each frame. Object_Labeling class cosists of variables Path_to_Labels, category_index, Vis_util.op_Labels and functions Vis_util.visualize_boxes,

label_on_image_array() which labels the each detected object in the frame. Text_to_Speech class consists of variables Engine, Rate, Volume and functions like Pyttsx.init(), engine.say(), engine.getProperty('rate'), engine.getProperty('volume'), engine.run.AndWait() which converts all the stored labels into speech.

- **Use Case Diagram**

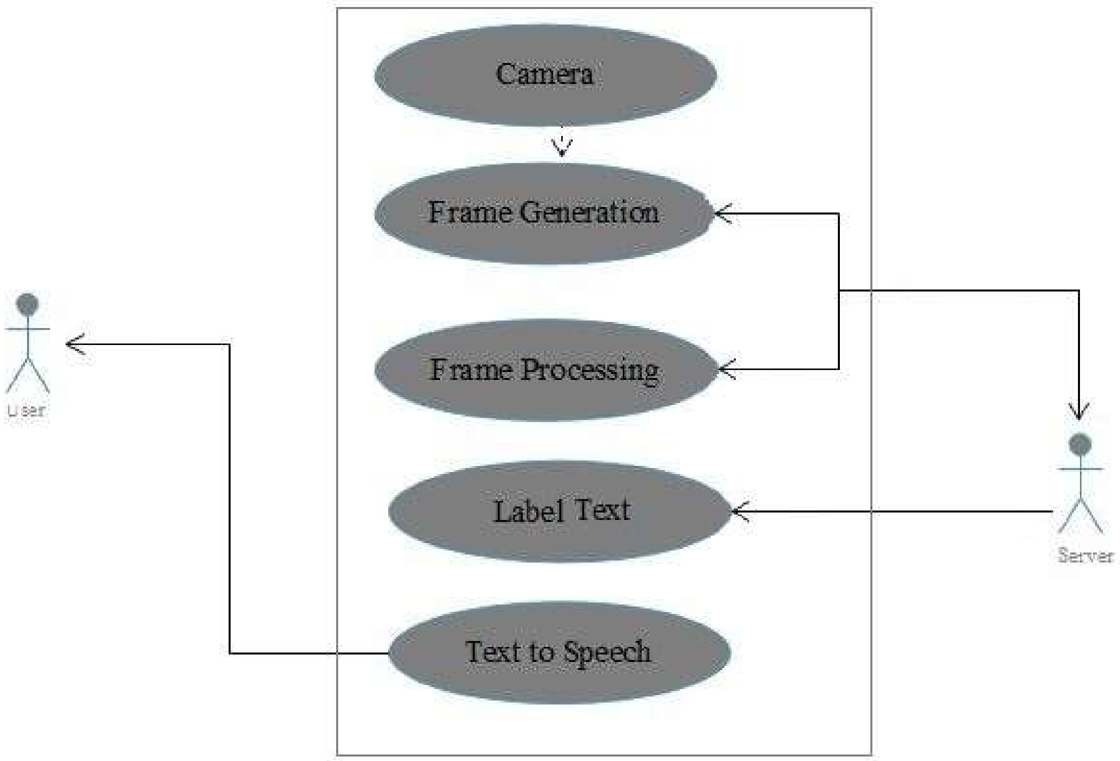


Fig 8.2.1.2 Sequence diagram For Smart Cap

- In this system, user doesn't interact with system directly. The interaction of user with the system is just an wearing the cap on which this system is implemented, turning on the system and at the end getting the final output generated by system.
- Here user wears the cap at the time when he requires an output. user turns on the system only once then system will work continuously till the system is turned off manually or technically. And user gets continuous output after the system starts working.

5.2.2 Dynamic modeling:

- **Data Flow Diagrams**

During the operation of system, general data flow is done as shown in DFD Level 0 diagram. Frame is given as input to the system for processing and as a output the speech is generated.

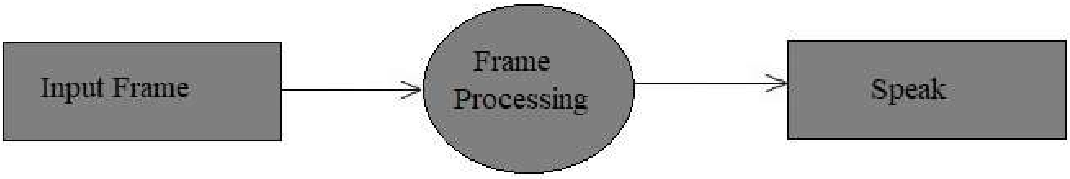


Fig 8.2.2.1 DFD level 0 for Smart cap

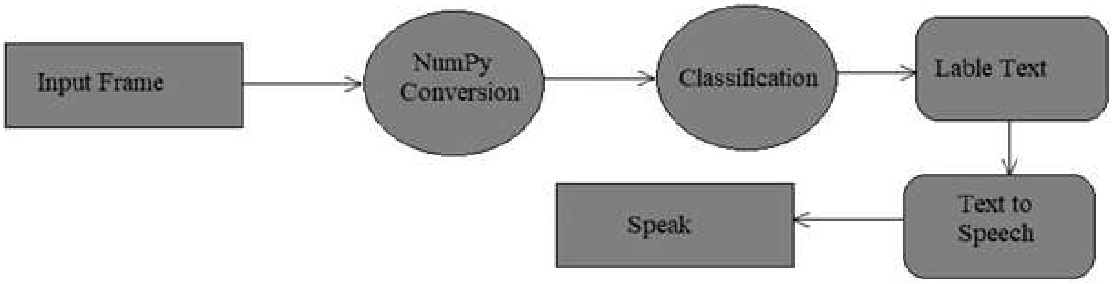


Fig 8.2.2.2 DFD level 1 for Smart cap

During the operation of system, actual data flow is done as shown in DFD Level 1 diagram. In this system, input is nothing but a frame captured by Camera which is passed to NumPy Conversion which convert the frame into numeric array which is used to detect the object. This array is passed to the Classification phase which matches the input array with the Object database library provided by COCOAPI dataset. After the object detection, system assigns the label to the particular object. Then system converts that labels into text i.e. Here full sentence is get generated which is going to converted into audio. This text to speech conversion is done by pyttsx3 library. At the end this audio sentence is provided to the speaker where user actually get the output.

5.2.3 Sequence Diagram:

When the system gets started, system immediately sends the message to the server through which server get notified that system is started and ready to work. Then server sends the frame request to the system. Then system captures the frame through the camera and sends back as an input to the server. Then server processes the frame, detect the objects, assigns the labels and converts that labels into text sentences. Then at the end server sends that text sentences to the system. Then system converts that text sentences into audio and speaks through the headset.

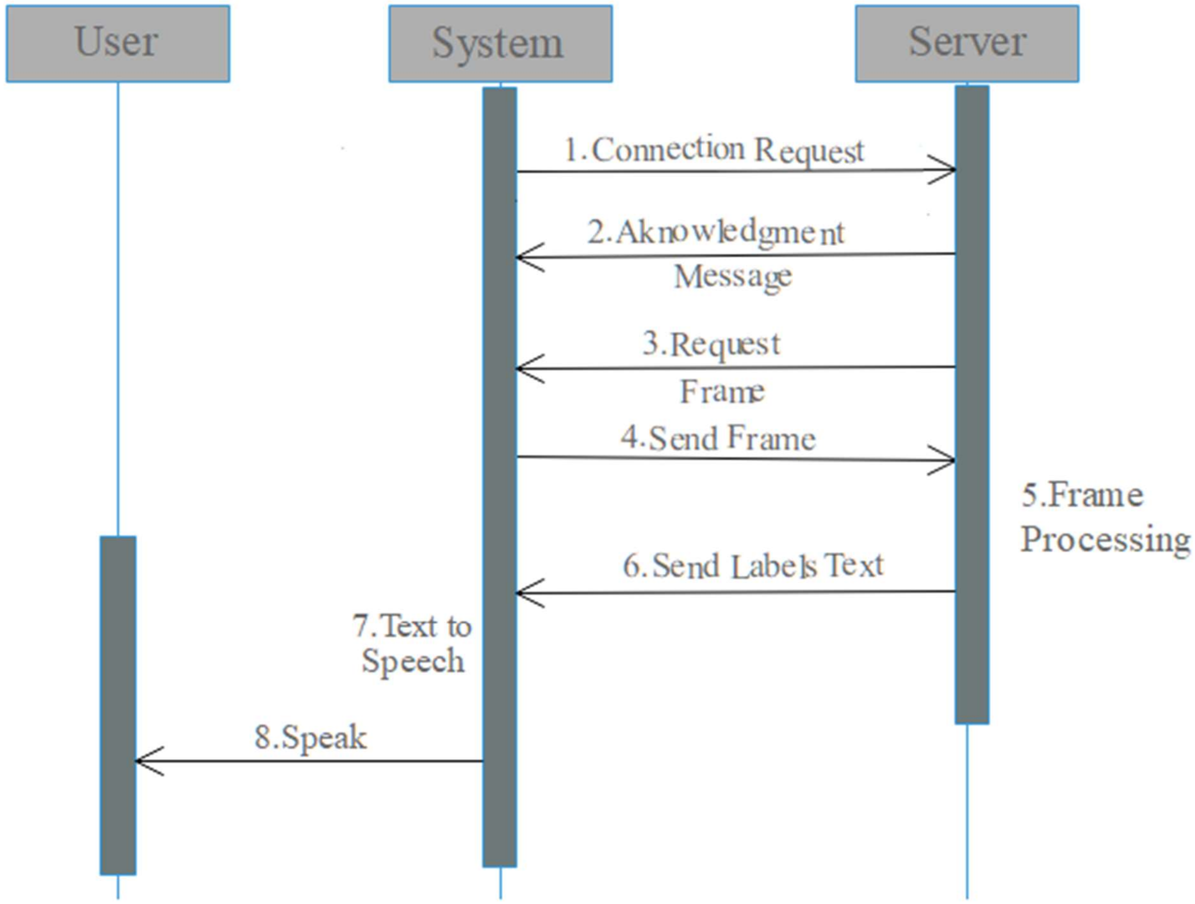


Fig 8.2.3.1 Sequence Diagram for Smart Cap

5.2.4 Activity Diagram:

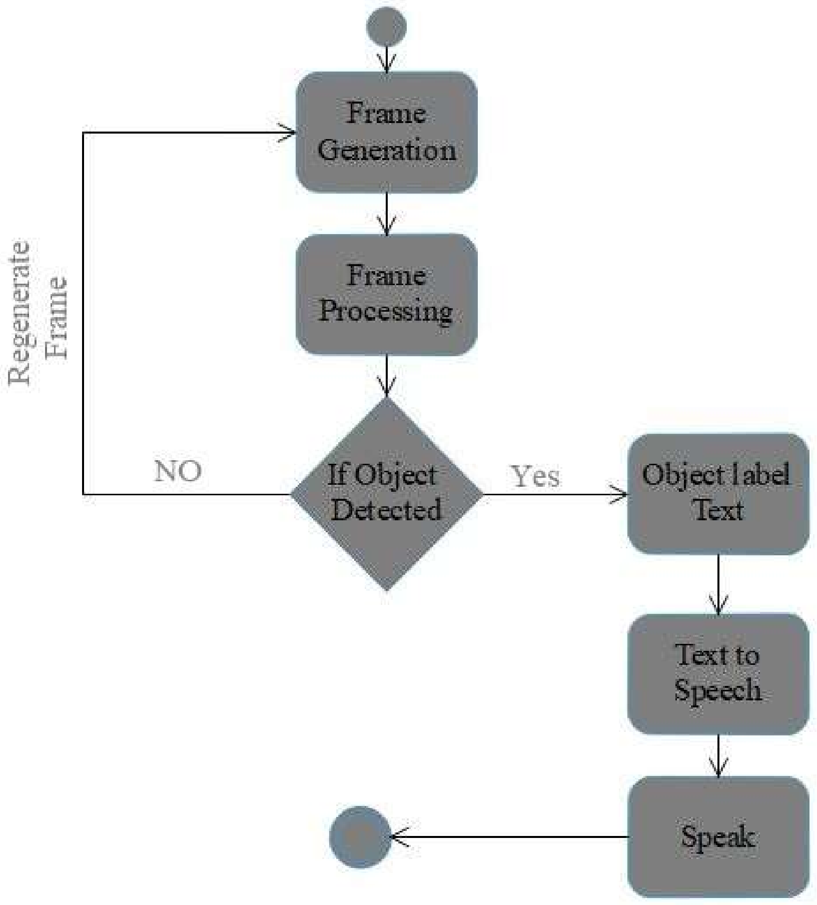


Fig 8.2.4.1 Activity Diagram for Smart Cap

- In above activity diagram first activity is Generation of frame using webcam and transferring it to the server.
- The second activity is processing a generated frame. If any object gets detected in that frame then labels are assigned else frame is regenerated.
- After labeling the object, labels are converted into text and passed to the system.
- Text is converted into speech and given as an output to the user.

5.2.5 Component Diagram:

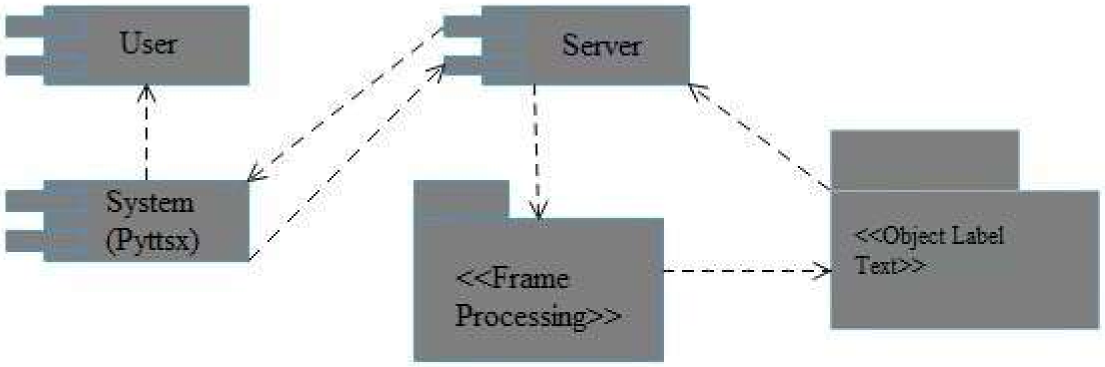


Fig 8.2.5.1 Component Diagram for Smart cap

- Above component diagram represent the basic requirement of system.
- The system contains hardware components like Creative Webcam, Raspberry Pi 3 and software components like pyttsx3 and Raspbian Buster.
- The server contains software components like TensorFlow and OpenCV for processing a frame and labelling the object detected in frame.

Chapter 6

Planning and Scheduling of project

6.1 Process Planning-

- Scheduling mainly concerns with minimum days required for completion each and every phase of project development. The Waterfall Model of software engineering is taken into consideration while doing planning, scheduling and development of the project.

Sr. No.	Phases of Project	Start Date	End Date	Days required for completion
1.	Requirement Gathering, Analysis	1 July,2019	30 July,2019	26 Days
2.	Design	1 August,2019	30 September,2019	46 Days
3.	Implementation	1 October,2019	15 January,2020	90 Days
4.	Testing	16 January,2020	28 February,2020	42 Days
5.	Final Execution and project report.	1 March,2020	1 April,2020	30 Days

6.2 Effort Estimation

- **Effort Estimation:**
  The effort which is taken by the project development team by considering all phases of project development life cycle is mainly calculated by using COCOMO Model.

- **Basic COCOMO Model:**

Software Project	**a_b**	**b_b**	**C_b**	**D_b**
1.Organic	2.4	1.05	2.5	0.38
2.Semi-detached	3.0	1.12	2.5	0.35
3.Embedded	3.6	1.20	2.5	0.32

The Formula of Estimation of Effort is:

$E= a_b \text{ KLOC } b_b$

$D= c_b \text{ E } d_b$

Where,

E= Effort Estimation.

KLOC= Thousand Lines of Code done by programmer.

a_b, b_b, c_b, d_b = Coefficients or Constants.

D=Development time.

For this embedded project,

$a_{b=}$ 3.6

b_b=1.20

KLOC= 700 LOC= 0.700 LOC

$E= a_b$ *KLOC^b_b

E= (3.6*0.70^1.20)

E= 2.34

Also,

D= c_b*E^d_b

D=2.5*6.48^0.32

D=1.91

Chapter 7

Risk Management and Analysis

The Risk Analysis means risk containment and mitigation. First, we have identified the risks and then planned. Then ready to act when a risk arises, drawing upon the experience and knowledge of the entire team to minimize the impact of risk on project.

Most software engineering project is inherently risky because of the variety potential problems that might arise. The software project risk may be due to new and unproven technologies, user and functional requirements, complex application and system architecture, performance and organizational issues.

Risk analysis includes following points: -

1) Identification of risk:

   Here, mostly the identification of source or root of the risk is done.

2) Classification of risk:

   There are various types of risk that user risk, software risk, hardware risk, network risk according to it risk has been get classify.

3) Plan for minimizing the risk:

   What should the methodology that the user has to follow for planning the risk minimization.

4) Implement mitigation action:

   What kind of proper work or action that should have to perform in terms of controlling the risk.

5) Communicate risk status throughout the project:

   What is the risk status at each and every stage of the project is get monitored and communicated to the project concerns people.

Project Risk Analysis:

Sr. No.	Risk	Identification of Risk	Classification of Risk	Plan for minimizing the Risk	Implement mitigation action	Communicate Risk status
1.	Damage of Raspberry pi	By checking raspberry pi signals	Hardware risk	Monitoring the use raspberry pi.	Do proper connection and interfacing	By monitoring status of the raspberry pi
2.	Improper use of system	Functioning of system is not properly working	User, software and hardware risk	Checking status of output given to the user.	Use system properly	By checking condition of output.
3.	Failure of Network	By checking network	Network risk	Checking network connection after interval of time	Provide alternate way of network	Checking network connection after interval of time
4	Training on Raspberry Pi	New device needs training	Hardware risk	Self-training	Utilization web resources for trainee	Able to handle Raspberry Pi and hardware efficiency

Chapter 8

References

Referred Papers

1] Paper Title: Challenges faced by students with blindness
Authors: Cuthbert Majoni and Julieth Mashatise

2] Paper Name: Effective Fast Response Smart Stick for Blind People.
Author Name: Ayat Nada, Samia Mashelly, Mahmoud A. Fakhr, and Ahmed F. Seddik

3] Paper Name: Computer Access for Persons Who Are Blind or Visually Impaired.
Author Name: I Environmental Research Institute of Michigan, Arlington, Virginia

4] Paper Name: Smart Glasses for the Visually Impaired People
Author Name: N. Sriskanthan, F. Tan, A. Karande

Web Reference

1] https://diyhacking.com/raspberry-pi-home-automation

2] https://www.raspberrypi.org/forums/viewtopic.php?f=37&t=42494

3] https://www.hackster.io/tushar-chugh/smart-cap-vision-for-the-visually-impaired-ca0ea1

4] https://www.livestrong.com/article/241936-challenges-that-blind-people-face/

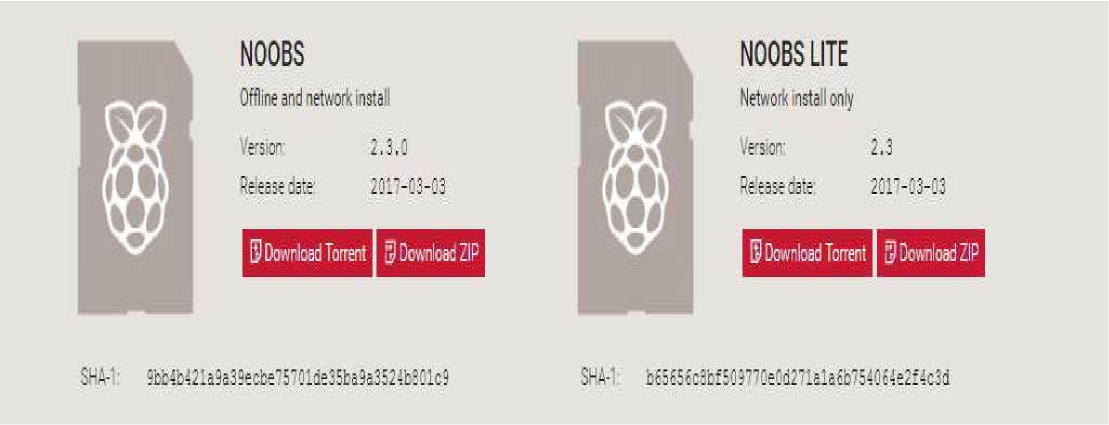
5] https://www.youtube.com/watch?v=fTvIIw3FfUg

6] https://www.cumulations.com/blogs/103/role-of-artificial-intelligence-in-iot

Annexure "A"

Installation of Raspbian on Raspberry-Pi

Step 1: Download NOOBS and extract it



NOOBS

Offline and network install

Version: 2.3.0

Release date: 2017-03-03

SHA-1: 9bb4b421a9a39ecbe75701de35ba9a3524b801c9

NOOBS LITE

Network install only

Version: 2.3

Release date: 2017-03-03

SHA-1: b65656c8bf509770e0d271a1a6b754064e2f4c3d

You're going to use your computer to put NOOBS on an SD card – so step one is to get NOOBS onto your computer!

Click here to head to the NOOBS download page.

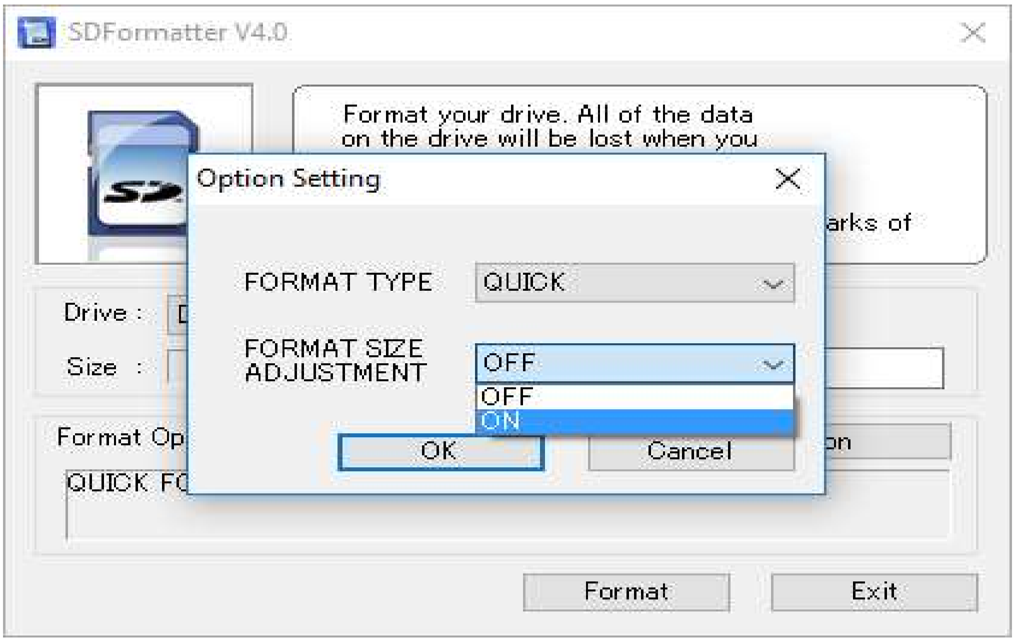
The NOOBS download page will let you choose between NOOBS and "NOOBS Lite." NOOBS includes a full version of Raspbian, so you can install that particular operating system without using the internet at all. With NOOBS Lite, on the other hand, you'll need a network connection to install any of the operating systems NOOBS makes available – even Raspbian.

Go ahead and choose whichever version you would like. NOOBS will download as a .zip file, so before you do anything else, go ahead and extract it.

Step 2: Format an SD card

Now you're going to want to go ahead and stick your SD card into the corresponding slot on your computer. You're going to want to format it as FAT. There are a few ways to do this:

On Mac or Windows, use the **SD Association's Formatting Tool** (Mac users can also just use the disk utility). Make sure the "Format size adjustment" option is set to "on." Then erase it in FAT (or MS-DOS) format.



SDFormatter V4.0

Format your drive. All of the data on the drive will be lost when you

arks of

Option Setting

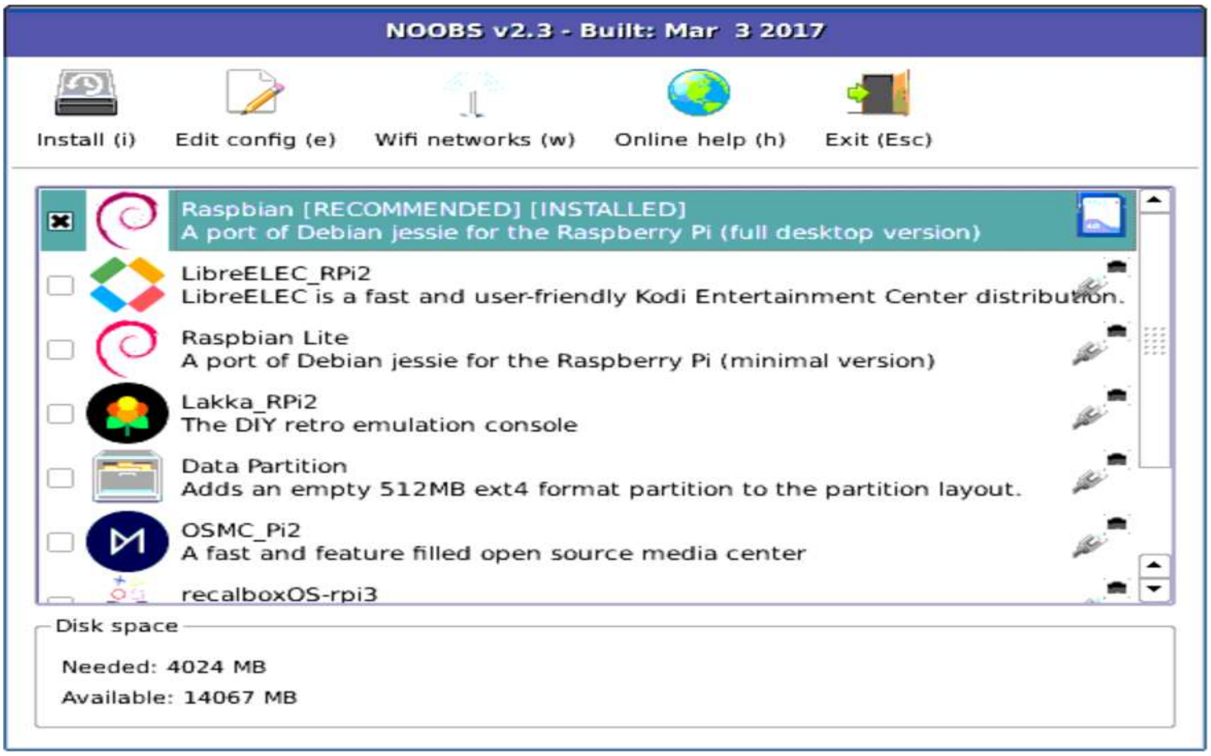
FORMAT TYPE: QUICK

FORMAT SIZE ADJUSTMENT: OFF

Step 3: Put the NOOBS files on the SD card

Now, just drag and drop the NOOBS files into your newly formatted SD card. You want the files only, so if your .zip extracted to a folder, open that folder up and select only the stuff inside of it.

Step 4: Put your SD card into your Raspberry Pi and boot it up



☒ Raspbian [RECOMMENDED] [INSTALLED]
A port of Debian jessie for the Raspberry Pi (full desktop version)

□ LibreELEC_RPi2
LibreELEC is a fast and user-friendly Kodi Entertainment Center distribution.

□ Raspbian Lite
A port of Debian jessie for the Raspberry Pi (minimal version)

□ Lakka_RPi2
The DIY retro emulation console

□ Data Partition
Adds an empty 512MB ext4 format partition to the partition layout.

□ OSMC_Pi2
A fast and feature filled open source media center

□ recalboxOS-rpi3

Disk space

Needed: 4024 MB
Available: 14067 MB

Once you have NOOBS on your SD card, using it is incredibly easy. Just put the SD card into your Raspberry Pi and start that sucker up. As we said before, while this guide is called “How to install NOOBS on the Raspberry Pi,” the endgame here is actually to install an operating system like Raspbian, LibreELEC, OSMC, or any of the others NOOBS gives you access to.

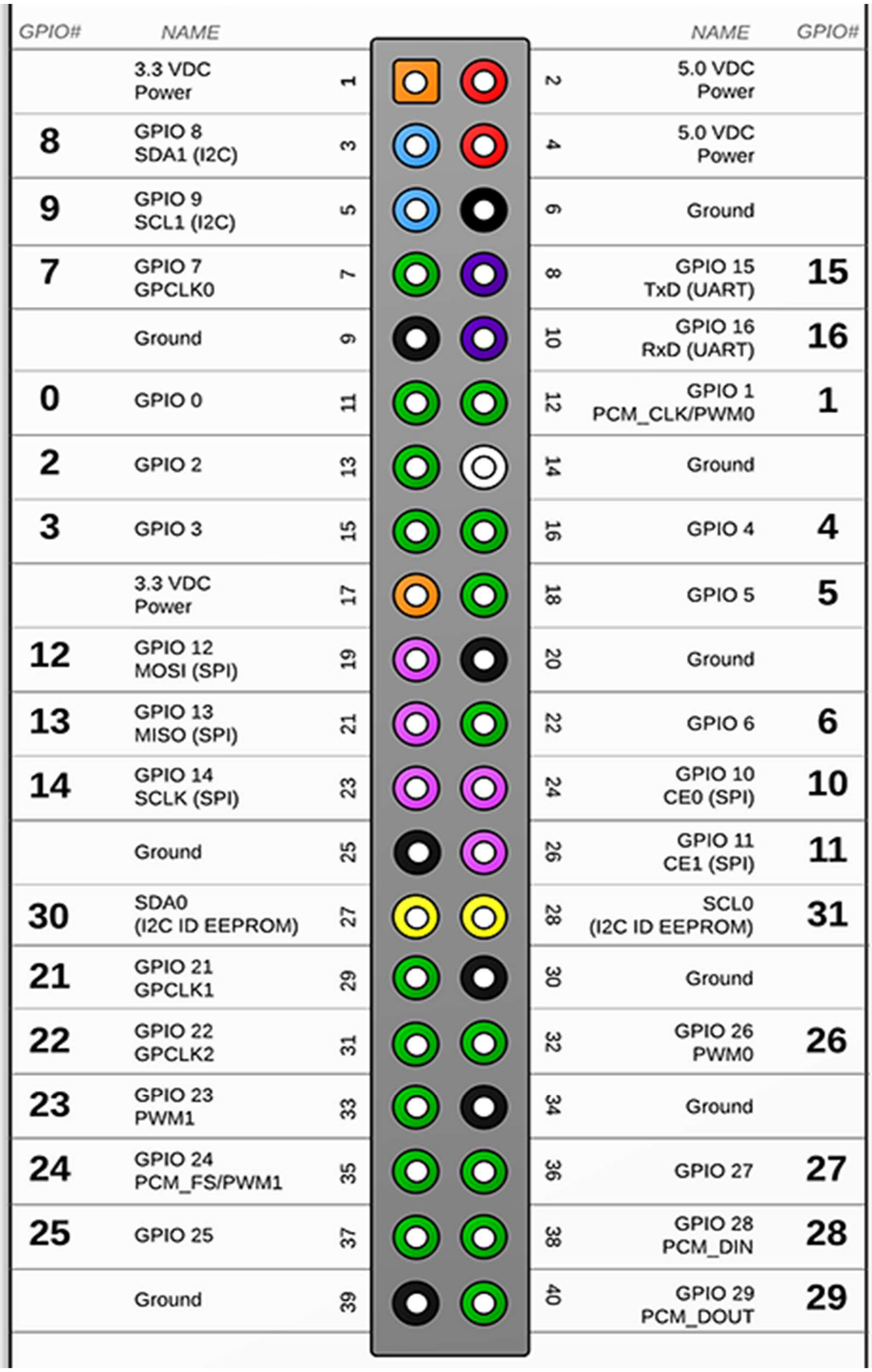
This is the step in which that happens. After booting to NOOBS, you’ll be greeted with a menu that will let you choose which operating system you’d like to install on your Pi. Your menu may look a little bit different than the one in the screenshot above, because NOOBS ingeniously adapts to your generation and model of Raspberry Pi.

Which OS should you choose? Well, that’s up to you. Raspbian is probably the most frequently used, and you’ll find plenty of projects here on our site that utilize it. OSMC acts as a media center, and LibreELEC boots directly to the popular media center app Kodi. Ultimately, it’s all a matter of personal preference!

Once you’ve decided, just hit “Install” and sit back. From now on, your Pi will boot directly to that operating system.

Annexure “B”

- **GPIO pin structure on Raspberry-Pi**



GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1	2	5.0 VDC Power	
8	GPIO 8 SDA1 (I2C)	3	4	5.0 VDC Power	
9	GPIO 9 SCL1 (I2C)	5	6	Ground	
7	GPIO 7 GPCLK0	7	8	GPIO 15 TxD (UART)	15
	Ground	9	10	GPIO 16 RxD (UART)	16
0	GPIO 0	11	12	GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13	14	Ground	
3	GPIO 3	15	16	GPIO 4	4
	3.3 VDC Power	17	18	GPIO 5	5
12	GPIO 12 MOSI (SPI)	19	20	Ground	
13	GPIO 13 MISO (SPI)	21	22	GPIO 6	6
14	GPIO 14 SCLK (SPI)	23	24	GPIO 10 CE0 (SPI)	10
	Ground	25	26	GPIO 11 CE1 (SPI)	11
30	SDA0 (I2C ID EEPROM)	27	28	SCL0 (I2C ID EEPROM)	31
21	GPIO 21 GPCLK1	29	30	Ground	
22	GPIO 22 GPCLK2	31	32	GPIO 26 PWM0	26
23	GPIO 23 PWM1	33	34	Ground	
24	GPIO 24 PCM_FS/PWM1	35	36	GPIO 27	27
25	GPIO 25	37	38	GPIO 28 PCM_DIN	28
	Ground	39	40	GPIO 29 PCM_DOUT	29

1.Power Pins

- The header provides 5V on Pin 2 and 3.3V on Pin 1. The 3.3V supply is limited to 50mA.

- The 5V supply draws current directly from your microUSB supply so can use whatever is left over after the board has taken its share. A 1A power supply could supply up to 300mA once the board has drawn 700mA.

2.Basic GPIO

- The header provides 17 Pins that can be configured as inputs and outputs. By default they are all configured as inputs except GPIO 14 & 15.

- In order to use these pins you must tell the system whether they are inputs or outputs. This can be achieved a number of ways and it depends on how you intend to control them. I intend on using Python.