A Raspberry pi Based Speaker Recognition System for Access Control

Nkolika O. Nwazor

Lecturer, Department of Electrical/Electronic Engineering, University of Port Harcourt, Rivers State, Nigeria

---***---

Abstract - *This work presents the design of an automatic speaker recognition(biometric) system using the raspberry pi as the control unit. Speaker recognition is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. This technique makes it possible to use the speaker's voice to verify their identity and control access to services such as voice dialing, banking by telephone, telephone shopping, database access services, information services, voice mail, security control for confidential information areas, and remote access to computers. Such a speaker recognition system has potential in many security applications. For example, users have to say a PIN (Personal Identification Number) in order to gain access to their office doors. By checking the voice characteristics of the input utterance, using an automatic speaker recognition system similar to the one that we will describe the system is able to add an extra level of security. The Bob spear algorithm is used in a python library to achieve speaker recognition on the raspberry pi.*

Key Words: Raspberry pi, Automatic, Speaker Recognition, Access Control, python, Bob Spear Algorithm

1. INTRODUCTION

In recent time, the issue of security and privacy has become of utmost importance. Government agencies, security agencies, and even private organizations seek to make their assets safer and to restrict access to private areas.

Biometric authentication is the verification of a user's identity by means of a physical trait or behavioral characteristic that can't easily be changed, such as a person's face" (Russel, 2005). This is the strength that the biometric systems have. In fact, the value of most part of the biometric properties of a person is assumed not to change for a long period, so after their storage, it is possible recognize someone through corporal and behavioral characteristics.

In forensic applications, the use of fingerprint identification can prove to be futile as it is quite easy these days to put on hand gloves when committing a crime or to forge another person's fingerprint in the bid to frame the individual.

Furthermore, it is quite difficult or almost impossible to distinguish between twins when using the face recognition approach for access control. This could lead to granting of access to wrong persons and compromise the security of the asset being guarded.

The use of voice biometrics for identification and recognition is more secure as it is capable of filtering out background noise and is not affected by temporary changes because up to 50 different characteristics are measured to create a voice print.

2. LITERATURE REVIEW

In the work (Sanchez, 2010), a speaker recognition tool for handheld computers was developed. The tool is written in C++ and it requires a single word to be used for the training phase and the same word for the recognition phase. It used the Euclidean distance to find the best match. (CMUSphinx, 2014) is a solution originally for speech recognition, but its modular design allows for extension in order to perform voice biometrics. Other solutions in Python such as voiceid (Google, 2011) and Speaker Recognition (SPEAR) (Python, 2012) also exist.

Given the fact that the task of voice biometrics consists of computationally expensive operations, such as feature extraction and creation of models, systems that utilize a server-based approach have been proposed. (Chowdhury, 2010) describes a distributed system for speaker recognition is which uses Gaussian Mixture Model Universal Background Model (GMM-UBM) models. {Brunet et al, 2013 }proposed a method that performs the entire speaker recognition process on an Android mobile device by extracting the MFCC features, and storing them as a distance vector. For the pattern-matching step, during the testing phase, they compare the test samples and extract their Euclidean distance. They used samples both from a publicly accessible and a private database, achieving promising accuracy results.

3. METHODOLOGY

The design of the Voice biometrics Module for Access Control, using the Raspberry Pi Single board computer, SBC is mostly software based. All the hardware components are ready-made peripherals, which are directly attached to the Raspberry Pi and configured for the specific purposes.

Basically, the project can be classified as an Internet of Things (IoT) project. It requires internet connectivity for it to function. This internet connectivity can be provided by a wireless dongle attached to the raspberry pi zero or a LAN cable or a USB 3G dongle. In this project, a USB Wi-Fi adapter is used to provide the raspberry pi zero with internet connectivity.

This project is made up of fundamentally two parts:

i. The Hardware Part
ii. The Software Part

The fig.1 shows the different blocks that make up the module.

3.1 The Hardware Module

Figure 1 shows the block diagram of the hardware module of the system.

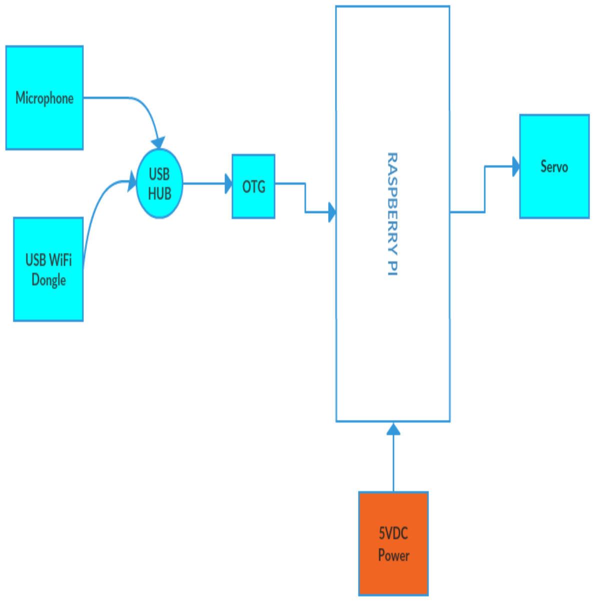


Fig-1: Block Diagram of Voice biometric system for Access Control

The different components that make up the module as follows:

1. Raspberry pi zero: The Raspberry Pi Zero is a full-fledged member of the Raspberry Pi family, featuring the following:
    a) A Broadcom BCM2835 application processor
    b) 1GHz ARM11 core (40% faster than Raspberry Pi 1)
    c) 512MB of LPDDR2 SDRAM
    d) A micro-SD card slot
    e) A mini-HDMI socket for 1080p60 video output
    f) Micro-USB sockets for data and power
    g) An unpopulated 40-pin GPIO header
    h) Identical pinout to Model A+/B+/2B
    i) An unpopulated composite video header
    j) Our smallest ever form factor, at 65mm x 30mm x 5mm

    Raspberry Pi Zero runs Raspbian OS, a Debian distribution. It also runs some Linux applications, including Scratch, Minecraft and Sonic Pi.

2. Wi-Fi Dongle: The USB Wi-Fi dongle is use to provide internet connectivity the raspberry pi zero. It is a low power device and it can be powered by the RPi Zero (Raspberry Pi Zero).

3. USB Mic: The USB Mic comes with an inbuilt sound card and is used as an input device for the voice recognition. It is low power and can run comfortably off the power supplied by the raspberry pi USB port.

4. USB Hub: The USB Hub is used to provide multiple USB inputs to a single port on the raspberry pi. The USB Hub used for this project is not at active type, i.e. is not a powered USB hub. The Wi-Fi dongle and the USB mic. are attached to the USB hub.

5. OTG Cable: The On-The- Go (OTG) cable is used as a converter for the micro USB port present on the raspberry pi to a normal port. The cable is also used as an extension for the peripheral devices attached to the RPi (Raspberry Pi).

6. 5VDC Power Adapter: A standard 5V 2A Power Adapter is used to power the raspberry pi zero.

7. Servo: The 5C servo motor is attached to the GPIO pins of the raspberry pi. This is used to simulate a door access when the speaker is recognized.

8. Memory card: This is the hardware which contains the OS (Operating System) running on the raspberry Pi. It is advisable to use at least an 8G Class10 Micro SD Card as the OS takes up to 4 gigabytes on installation. The Class 10 stands for the read and write speed of the memory card.

As the power adapter supplies the raspberry pi zero with power. The servo in turn takes its power from the raspberry pi's 5V Voltage output supply pin. As a pre-registered command is spoken and the system identifies the speaker, the program is written in such a way that it send a pulse width modulated signal through the GPIO pin in which the servo's signal pin is connected to. This causes the servo to rotate.

Fig. 2 is the circuit schematic of all the peripheral devices used with the raspberry pi zero:

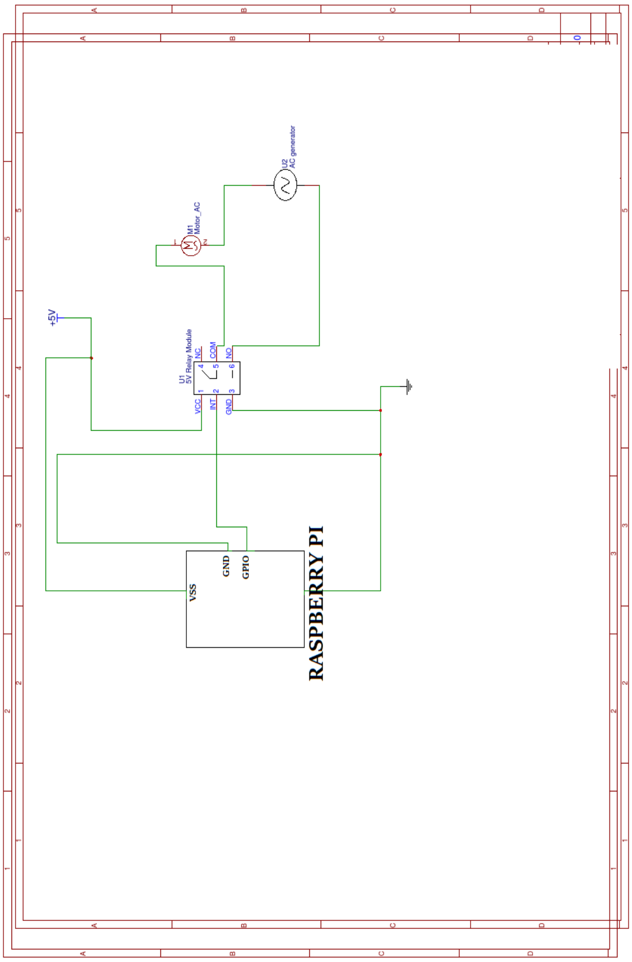


Figure 2: Circuit Schematic of the Speaker Authentication System

3.2 Software Module

The software used to accomplish the objective of this project are as follows:

i. **The Raspbian Jessie Pixel:** This is a Linux distribution which is more of a combination of Debian and Raspberry. It is built specifically to run on the Raspberry Pi. It has a desktop view and also a command line interface.

ii. **The SD Card Formatter:** This is a Windows-based software which is used to format the microSD card in which the Raspbian image is going to be mounted on.

iii. **Etcher:** This is the software used to write the Raspbian image to the Raspberry Pi microSD card.

iv. **Putty:** Putty is a tool that enables one to connect to the command line interface (CLI) of the raspberry pi through a protocol known as SSH (Secure Shell). The raspberry pi and the computer that request access to the raspberry are required to be connected on the same Local Area Network (LAN). Also, the username and password of the raspberry pi to be accessed is needed when trying to login to the raspberry through SSH, using Putty.

v. **VNC Viewer:** This is a Windows-based application that provides a server for viewing the X windows of the raspberry pi

vi. **Google Assistant SDK:** The google assistant SDK is a tool kit which contains functions and parameters which were used to build the speech to text and text to speech functionality. After the Speaker has been identified by the module, what the module uses to understand exactly what is being said is the Google assistant SDK. It is a cloud based tool and therefore requires internet connectivity to be actively ON in order for it to function.

vii. **The Piwho Library:** This is an open source Python library which has been employed in this project to achieve Voice biometrics.

viii. **GPIO library:** This is an open source Python based library that is used to address the GPIO (General Purpose Input Output) pins of the raspberry pi zero

3.3 Design Steps

The process of achieving the voice biometrics for access control using the raspberry pi has been broken down into steps as follows:

i. The Raspbian OS image is downloaded
ii. The SD Card formatter ids used to format the microSD Card in order to prepare it for the mounting of the Raspbian OS image.
iii. The Etcher Software is used to burn the image file to the microSD Card
iv. After the image has been burnt into the microSD Card, the it is then mounted into the Raspberry Pi Zero and the Pi Zero is powered up.
v. All the peripherals are then attached to the USB hub
vi. After the raspberry pi zero is connected to a wireless network, the firmware is updated and upgraded.
vii. The Piwho library is downloaded from gitHub
viii. All the source files are built using the Cmake tool
ix. Thereafter, the training phase is launched. Here a user records are many short audio samples of his/her voice and saves it to a folder.
x. The folder is then used to train the piwho software and various parameters are assigned to the .xml file which is generated. This .xml file contains all the features of the voice which has just been trained
xi. The enrolment phase is complete
xii. An account is created with google in order to gain access to the google assistant SDK
xiii. The google assistant SDK is then downloaded and a python script is run to install it on the raspberry pi zero
xiv. Finally, a python script is written to integrate the whole system and trigger the servo when a registered user issues a command it recognizes.

Fig.3 is the flow chart of the python script written for this purpose

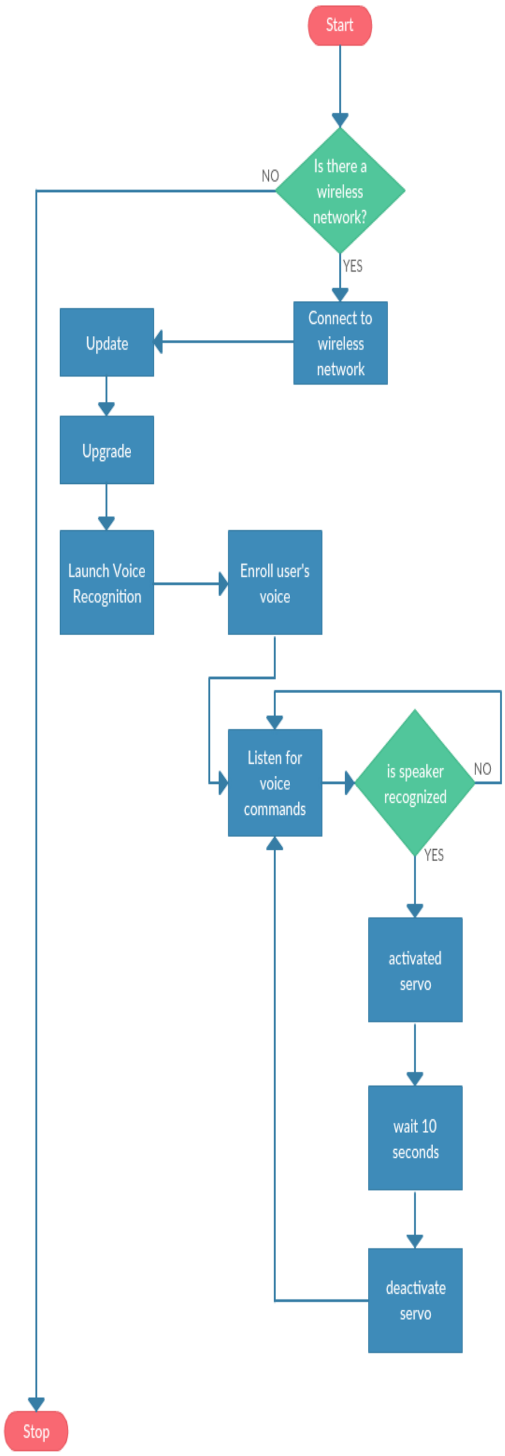


Fig. 3 - Flowchart for Voice biometrics for Access Control System

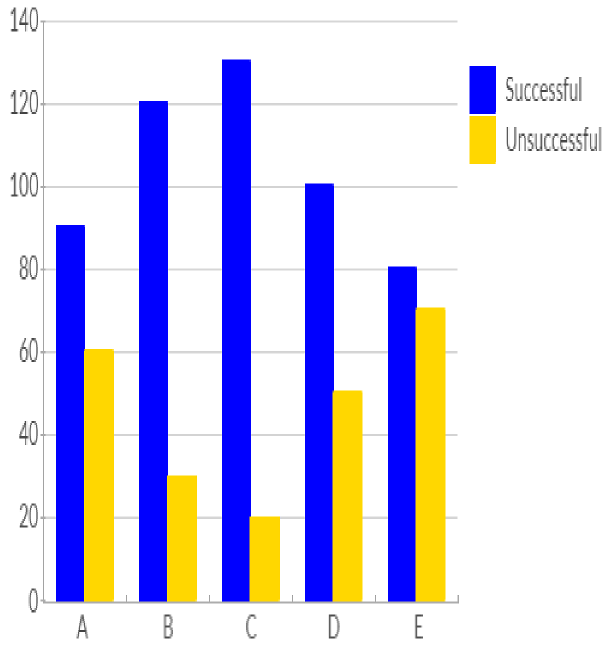
4. RESULTS

A prototype of the proposed system was developed. Experiments were performed to test the proposed system and to measure the accuracy of the system. Different audio samples (in .wav format) were taken and used to train the speaker recognizer. Thereafter, the voice of the same person which was enrolled was used in different conditions to test the accuracy of the system.

The various conditions by which the speaker recognition system was tested include:

i. Recognition in a crowded place with a lot of background noise
ii. Recognition in a silent place with little or no background noise
iii. Recognition when the speaker's voice is not loud
iv. Recognition when the speaker's voice is loud
v. Recognition when the speaker's voice has slight variations due to cold

The results obtained are as shown in fig. 4.



Same Speaker under Different Conditions Fig. 4 - Result of Tests Carried out on Same Speaker under

A = Recognition in a crowded place with a lot of background noise
B = Recognition in a silent place with little or no background noise
C = Recognition when the speaker's voice is not loud
D = Recognition when the speaker's voice is loud
E = Recognition when the speaker's voice has slight variations due to cold

Figure 4 represents the results of how well the module could perform on 150 test samples of the speaker's voice, after the speaker's voice has been pre-enrolled into the database of the module. Table1 shows these statistics:

Table.1: Success Rate of Speaker Recognition System

S/No.	Test Condition	Number of Accuracy	Percentage (%)
1.	Recognition in a crowded place with a lot of background noise	90/60	60.00
2.	Recognition in a silent place with little or no background noise	120/30	80.00
3.	Recognition when the speaker's voice is not loud	130/20	86.67
4.	Recognition when the speaker's voice is loud	100/50	66.67
5.	Recognition when the speaker's voice has slight variations due to cold	80/70	53.33

The success rate in percentage is shown in fig.5:

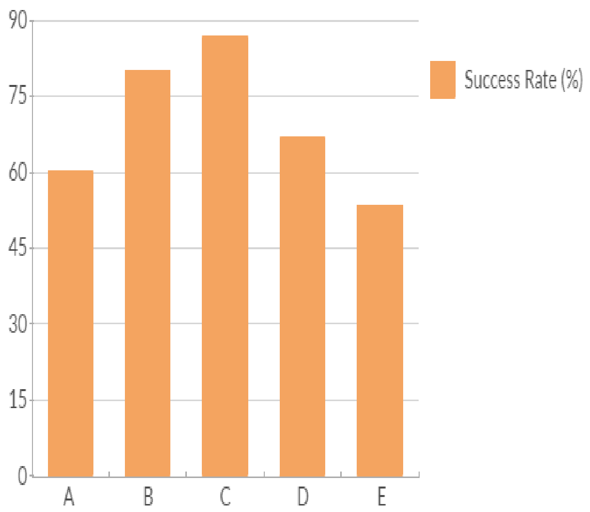


Fig.5: Success Rate of Speaker Recognition Module (%)

5. CONCLUSION

Speaker recognition for access control has been successfully implemented on the Raspberry Pi Single Board Computer. This system can be further industrialized and made to work in companies as their means of entrance to only authorized places in the firm. Furthermore, it can be used to securely log the data of the persons that have had an access to a particular place. From the tests and the results carried out, it can be deduced that the speaker recognition module performs best when the speakers voice is loud and when there is silence in the place where the module is performing computation. The response time of this module is relatively fast.

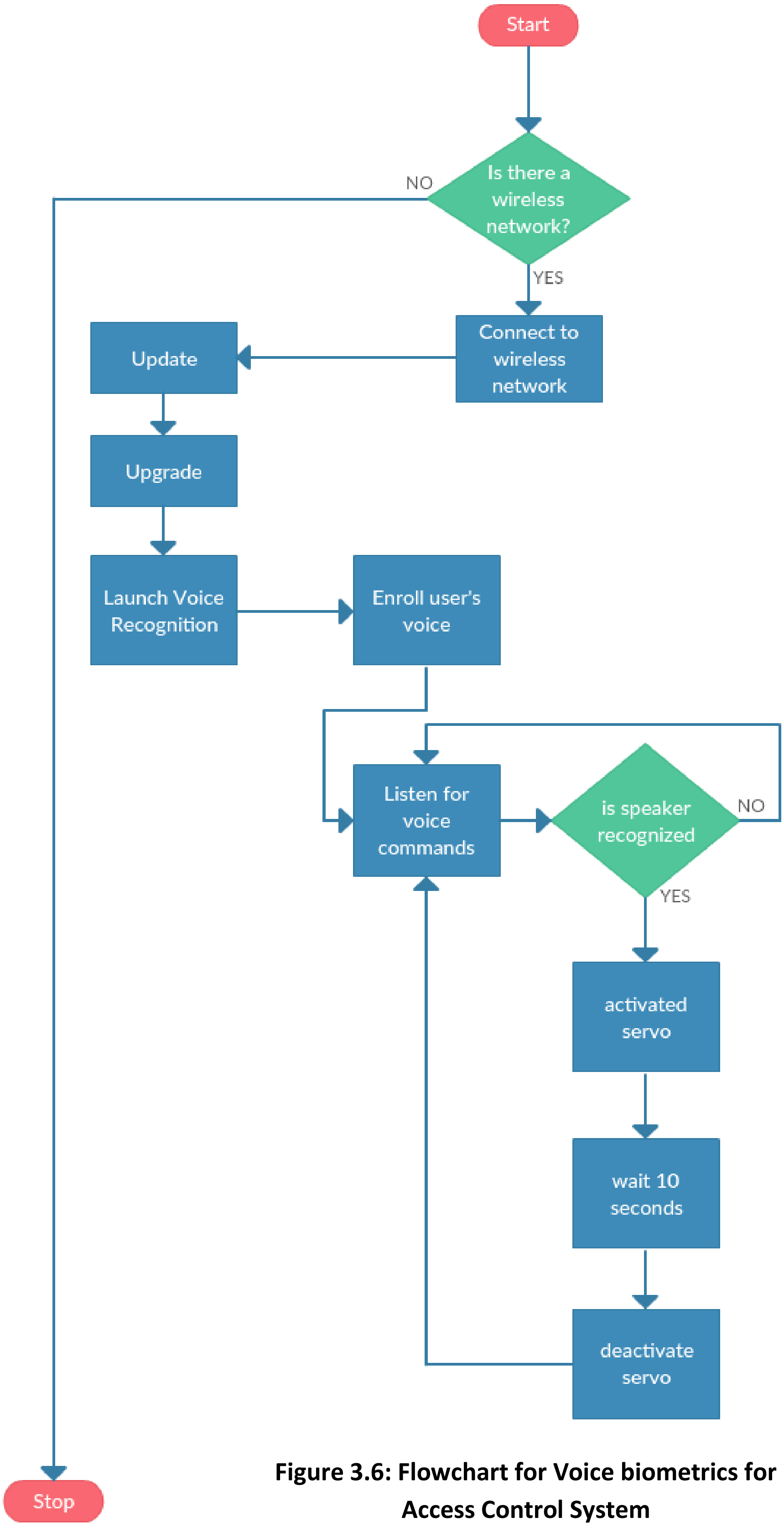


Figure 3.6: Flowchart for Voice biometrics for Access Control System

CHAPTER 4 IMPLEMENTATION AND TESTING

4.1 Tests

Experiments have been performed to test the proposed system and to measure the accuracy of the system. Different audio samples (in .wav format) were taken and used to train the speaker recognizer. Thereafter, the voice of the same person which was enrolled was used in different conditions to test the accuracy of the system. The various conditions by which the speaker recognition system was tested include:

vi. Recognition in a crowded place with a lot of background noise
vii. Recognition in a silent place with little or no background noise
viii. Recognition when the speaker's voice is not loud
ix. Recognition when the speaker's voice is loud
x. Recognition when the speaker's voice has slight variations due to cold

4.2 Results

The results obtained are as follows:
A = Recognition in a crowded place with a lot of background noise
B = Recognition in a silent place with little or no background noise
C = Recognition when the speaker's voice is not loud
D = Recognition when the speaker's voice is loud
E = Recognition when the speaker's voice has slight variations due to cold

The

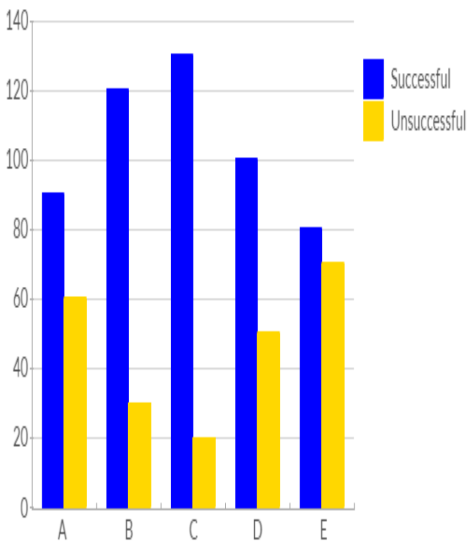


Figure 4.1: Result of Tests Carried out on

figure 4.1 represents the results of how well the module could perform on 150 test samples of the speaker's voice, after the speaker's voice has been pre-enrolled into the database of the module.
Table 4.1 Shows these statistics:

Table 4.1: Success Rate of Speaker Recognition System

S/No.	Test Condition	Number of Accuracy	Percentage (%)
1.	Recognition in a crowded place with a lot of background noise	90/60	60.00
2.	Recognition in a silent place with little or no background noise	120/30	80.00
3.	Recognition when the speaker's voice is not loud	130/20	86.67
4.	Recognition when the speaker's voice is loud	100/50	66.67
5.	Recognition when the speaker's voice has slight variations due to cold	80/70	53.33

Representing the success rate in percentage, we have the figure 4.2:

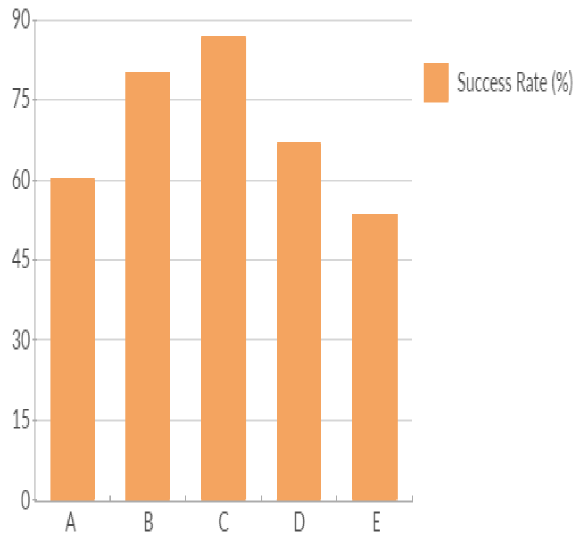


Figure 4.2: Success Rate of Speaker Recognition Module (%)

REFERENCES

D. Kornack and P. Rakic, “Cell Proliferation without Neurogenesis in Adult Primate Neocortex,” Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/science.1065467.

M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

K. Elissa, “Title of paper if known,” unpublished.