CS 370 Term Project

Jonathan Ibarra, Dinesh Giri, Erik Cervantes

Department of Computer Science, Colorado State University

CS 370: Operating Systems

Instructor Name

May 03, 2023

CS 370 Term Project

Single-board computers are inexpensive, and the Raspberry Pi is one of them. It was created with education in mind. For students who are interested in learning about computer science, programming, and electronics, it is a great resource. From straightforward LED displays to intricate robotics, a wide range of projects have utilized the Raspberry Pi. For our project, we will use a Raspberry Pi to build a door camera and voice recognition process. If a password is spoken, the Raspberry Pi will send an email notification displaying whether the password was correct or not.

Problem Characterization

When determining a project surrounding the Raspberry Pi, we spent hours of brainstorming. We wanted to make sure to create a system that did detection and activation. We first produced 2 ideas. Our first idea was to create a weather detection system. The Raspberry Pi would detect the weather and send an alert to a user's phone letting them know what to wear for the day. The issue we came across with that idea is we were unsure of a weather detection system. We then thought of a chicken feeder. The Raspberry Pi would be set up with a camera and weight sensor. The weight sensor would detect whether the chicken feeder is empty and activate a motor that would add food. The only problem with that idea is we did not have the weight sensor, nor the motor parts for the contraption.

The two parts that we had on hand were a camera and a microphone. With time not on our side, these were a constraint to what our project had to revolve around. After much deliberation, we decided to create a home security system. A camera would be on and detect facial recognition; once a face is detected, the microphone would be activated on a timer to detect a passcode. A notification would be sent to an email on whether the correct passcode was said.

Method

In this section, we go over the hardware and libraries used and procedure of how we went about our project.

Hardware

- Raspberry Pi 4B

Popular single-board computers like the Raspberry Pi 4 B are utilized for many different projects. It is appropriate for a variety of applications because it has a quad-core ARM Cortex-A72 CPU, up to 8GB of RAM, and a selection of input/output ports. Your project will increase the security of smart homes by constructing a door camera and sensor system using the Raspberry Pi 4 B.

- USB Web Camera

The Raspberry Pi 4 B can be connected to the camera module, which is a tiny, light camera, using a ribbon connection. High-quality photos and films may be taken via the camera, which is ideal for security applications. The camera is used in our project to take pictures of and record videos.

- USB Microphone

Libraries

- Cv2

A library of programming functions with a focus on real-time computer vision is called OpenCV (Open-Source Computer Vision Library). It is used to process photos and videos, including reading and writing image files, transforming images with filters, finding and tracking objects in images and videos, and many other activities.

- Time

Python's time module offers a variety of time-related methods, including calculating a program's elapsed time, sleeping for a predetermined amount of time, and translating between multiple time representations.

- Sys

The sys module gives users access to some variables that the Python interpreter uses or maintains as well as functions that have a close relationship to the interpreter, like routines for quitting the interpreter or obtaining command-line arguments.

- Speech_recognition as sr

Python uses the speech_recognition package to do voice recognition. It supports a number of speech recognition programs, including Sphinx, Microsoft Bing Voice Recognition, and Google Speech Recognition.

- Pyaudio

Python can record and playback audio thanks to the pyaudio library's interface with the computer's audio hardware.

- Smtplib

Python's Simple Mail Transfer Protocol (SMTP) is used to send emails, and the smtplib package is utilized for this.

- Pyttsx3

A Python package used for text-to-speech conversion is called pyttsx3. It can be used to read out a file's contents or to synthesize voice from text. Here is a quick guide on how to incorporate PyTTSX3 into our project.

Procedure

The first step we took to complete our project was to create facial recognition software. We got a lot of support from the OpenCV (Open-Source Computer Vision) library and code provided through their documentation. We wanted to get this part to work first so we could implement a condition statement when a face was detected. When a face was detected, we would use PyAudio to turn on the microphone and record any spoken words for a certain amount of time. While the microphone is turned on, the Speech Recognition software would check if a any words that were spoken matched to a predefined password. If there is a match, then the next step of the process to send a notification through email would begin. We utilized smtplib to send an email alert to one of our emails displaying whether the passcode matched or not.

Results

Working with the Raspberry Pi had its ups and downs, and getting the different components of the process was more than difficult. Our group also had to deal with projects and exams of other classes on top of this project. Getting the Raspberry Pi booted up and working took some time as we needed the correct HDMI adapter to display on a monitor. We couldn't get hold of the adapter, so we migrated to one of our group member's TV. Before we could do anything, we first needed to download a coding environment and python extensions; we used VSCode to do this. We then went on with our plan.

However, we discovered that the microphone was malfunctioning during testing because we were unable to record any audio. We double-checked the audio speaker's connections to the Raspberry Pi to make sure they were secure in order to troubleshoot this problem. We were mainly having difficulty with getting out USB microphone to be compatible with PyAudio's software and it just wasn't working.

The speech recognition component was then implemented using the PyAudio and Speech Recognition modules. The application was unable to accurately transcribe our speech due to problems with the microphone not working. We experimented with changing the settings and testing in various scenarios, but the speech recognition feature was still giving us trouble.

We then tried to use the smtplib library to develop the email alert feature. We wanted to at least have a feature that was activated when a face was detected. The usage of less secure apps is no longer permitted by Google, thus we discovered that we were unable to send emails using our Gmail account. After switching to a different email service provider, we were able to use our script to send emails without incident.

After much failure, we pivoted our project to then send a text message when a face is first seen in the camera frame. We utilize Twilio to be able to do this. We created a Twilio account, set up the client, and sent the message. This was to at least have an activation piece to our project.

Discussion

There were a few unavoidable issues we had to deal with when attempting to complete our project. The microphone not being able to function was a huge setback that were unfortunately unable to get through. With the microphone out of the equation, we tried to at least get the emailing part to work, but that also did not work unfortunately. A lot of problems were piling up that we couldn't figure out how to solve. We at least got a message to send, which combined with the facial recognition, is the heart of the final result. This may or may not have been avoided if we had begun our project any sooner than we did, although, as I've mentioned, the workload of all of our group members could not be avoided, therefore the effort that we put in was appreciated by all the group members. If things could have gone differently, we would have learned the ins and outs of the Raspberry Pi to better our programs' functionality and compatibility with the libraries we wanted to use.