Z-Recognition
TPJ665 Capstone
Project Report

Patrick Ziajski
Zarak Khattak

November 8, 2019

SENECA COLLEGE OF APPLIED ARTS AND TECHNOLOGY
School of Electronics and Mechanical Engineering Technology
1750 Finch Ave East, Toronto, Ontario, M2J 2X5. Tel. 416-491-5050
www.senecacollege.ca

1 Executive Summary

Z-Recognition is a License Plate Recognition system which provides clients with a new and intuitive way to implement and enforce parking systems. Our goal is to allow for the successful implementation of both hardware and software elements to allow for a working project. The hardware aspect of the project involves the implementation of a microcontroller which holds all the pre-required software to run the system. A camera that is used to capture images of License plates. A speaker and a Servo Motor.

The software aspect of the project entails the creation of a python script which brings together all the hardware elements and have them operate with each other. The script captures an image which is processed as a JavaScript Object Notation (JSON) using Microsoft Azure Computer Vision to process the image. The captured data is then cross referenced against a database for validity and the Servo Motor operates in response to valid data being found, whereas, the speaker operates if no matching data is found.

Contents

List of Figures

List of Tables

2 Introduction

Security is an important feature for a company providing a service to their customer. The company must ensure that its service can only be used by authorized personnel. This is where Z-Recognition comes into play. Z-Recognition is a license plate recognition system that ensures only authorized personnel have access to a designated area. Z-Recognition accomplishes this by taking an image of the approaching vehicle and processing it. If the vehicle is authorized, meaning the license plate is recognized, the vehicle is given access. Our group has decided on a license plate recognition system due to its uses in modern day society. Today, one can still find areas that are monitored by a single employee, or by a ticket-based entry system. These systems could be found inferior because of their reliance on periodic human interaction or supervision. Z-Recognition is designed to run autonomously with minimal costs, and a single requirement of an active internet. This license plate recognition system also implements a core feature in the future of computer technology, machine learning. With machine learning, an artificial intelligence (AI) can provide systems the ability to learn and improve without being explicitly programmed. This means that automated systems become more secure and reliable, without the need of constant supervision. Finally, this project requires both hardware and software implementation to be fully functional. It requires us to work with and learn both software development and hardware assembly which we, as Computer Engineering and Technology students, would prefer.

3 Functional Features

- Image processing - Using Microsoft's Cloud based Computer Vision Service, called Azure, a captured image will be uploaded to Azure's Service. The service perform Optical Character Recognition (OCR) on the image, and it will return a JavaScript Object Notation (JSON) object with all the recognized characters.
- Motor Control - The Raspberry Pi 3 Model B, a MicroController Unit (MCU), will control a servo-motor to imitate a gate arm used to allow entrance into a secure area.
- Scheduled script execution - The main python script will be executed at specified time intervals
- Sound execution - The Raspberry Pi will emit a sound from a speaker in order to audibly notify the personnel using the system of whether they have been given or denied access.

4 System Specifications

Camera

- 720p/30fps
- Fixed focus
- Field of View - 60°

Raspberry Pi 3 Model B

- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
- 1GB RAM
- BCM43438 wireless LAN
- 40-pin extended GPIO
- 4 Pole stereo output
- Micro SD

Servo-motor

- Operating voltage - 3.0V 7.2V

Microsoft Azure

- Active subscription
- Cognitive Services resource

5 Operating Instructions

Please follow the listed steps to ensure that the system is operating as intended:

- Ensure that the Raspberry Pi Microprocessor is powered on
  - Plug the power cord into the Pi and connect to a power source
- Connect the Pi to an internet source
  - Using an Ethernet Cable or connecting to a Wireless-Fidelity(Wi-Fi) access point
- Ensure that the Camera is connected properly to the Raspberry Pi via Universal Serial Bus (USB)
- Start the Main Python Script named "ZRecognition.py"

6 Product Design, Implementation, and Operation of the System

6.1 System Block-Diagram

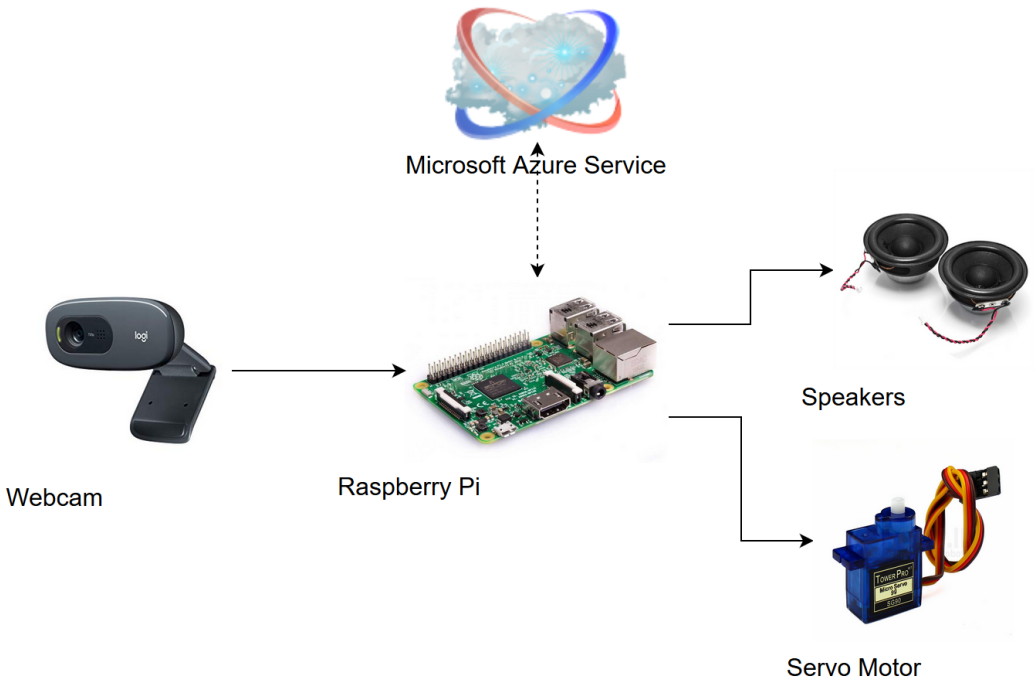

Figure 1: System Block-Diagram

6.2 Software Flow-Chart

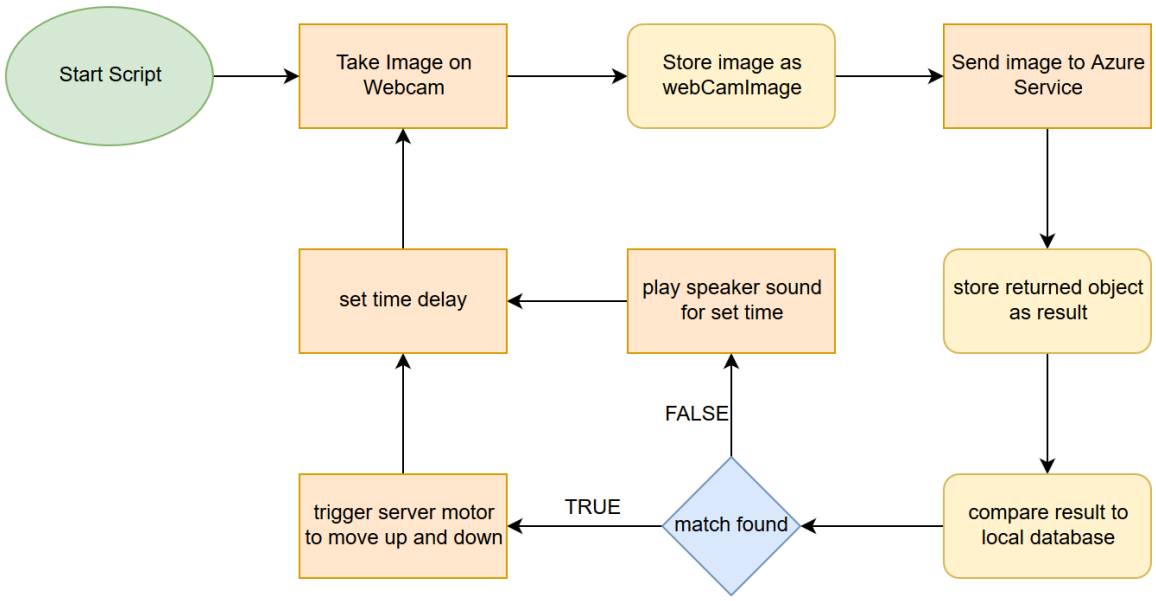

Figure 2: Software Flow-Chart

6.3 Components Description

The system utilized in this project consists of a few major components. Each component is listed and described below:

- Microprocessor - Raspberry Pi 3 Model B
  - The Raspberry Pi 3 Model B is the sole microprocessor for this project. It contains a python script that is executed to run the program. The Pi makes use of a 1.2GHz Quad Core 64bit CPU with 1GB built in RAM.
- Camera - Logitech C270 HD Webcam
  - The Logitech C270 Webcam is installed to ensure that the images captured are of high quality so that they are easy to interprit. These Images are taken in 720p with a fixed focus and field of view of 60° These images are interpreted by the Microsoft Azure Cognitive services and returned back as a JSON object.
- Servo Motor
  - The RioRand SG90 Micro 9d Servo motor simulates a gate opening and closing once the image data has been authenticated against a database of existing and valid license plates. The Motor operates utilizing a voltage range of 3.0V to 7.2V

6.4 Theory of operation

The system can be broken down into two different parts, one being hardware, the other software.

The hardware component of the project consists of a Raspberry Pi 3 Microprocessor, a Camera, a Speaker and a Servo Motor. The Pi contains on it a python script, which, when executed triggers the camera to take periodic image captures.

When the camera captures a License Plate Image, it dispatches the image to Microsoft Azure Computer Vision Resource. The Image is then deciphered and converted into a text block and returned back as a JSON Object. The JSON object is then cross referenced with a database of existing and authorized License Plates.

If the data exists in the database and there is a match, the Servo Motor is activated and rotates 90° to simulate a gate arm opening.
If the data does not exist in the database and there is no match, the Servo Motor does not activate and a sound is played through the speaker to alert the vehicle personnel that the license plate could not be validated.

7 Maintenance Requirements

Routine maintenance is a good way to avoid any unwanted interruptions for the services being provided. It is recommended that proactive maintenance be done on the system at a frequency of one week (7 days) or anytime that the system fails to operate.

- Reset the Raspberry pin (disconnect and reconnect power)
- Relaunch the Python script

Please ensure that you test the service after completing maintenance to ensure that all systems are up and running.
If you are unsure of how to complete any of these tasks, please get in touch with the contacts available in the contact(s) section of the report.

8 Conclusion

Z-Recognition provides a more modern, effecient, scalable and simplistic approach of a security system. The design and implemention of this system required aspects of a majority of the courses taken in Seneca College's Computer Engineering Technology program, primarily from ETD 555, LNX 155, NSP 655, PRG 469, and PRG 655. However, the scope of this project required research above the foundations learned in these courses. For example, setting up and using the Azure Computer Vision resource which required knowledge of JSON, Machine Learning, image processing, and more indepth knowledge of the HyperText Transfer Protocol (HTTP). This project was successfully completed, within the initial projected timeline previously defined.

9 Further Developments

As stated previously, this project is highly scalable. That means with very slight modifications to the project, it could be implemented in a wide range of situations, from a one car garage, to an airport parking lot with multiple entrances and exits for example. Future extentions to this project

- Motion detection for triggering the image capture and processing
- Low Light detection / Night vision for improved image capture
- Local computer vision service for a more reliable and independant system

A Parts List / Bill Of Materials

BILL OF MATERIALS		
Component Name	**Quantity**	**Price (CAD)**
Raspberry Pi 3 Model B	1	$35.00
Logitech C270 HD Webcam	1	$32.00
RioRand 5PCS x SG90 Micro 9d Servo	1	$16.99
Gikfun 2" 8 Ohm 2W Audio Speaker	1	$16.88
Total Cost w/ Tax		$113.98

Table 1: Bill of Materials

NOTE All prices are in Canadian Dollars (CAD)

B Credentials

Currently there are no log in credentials required for any aspect of the software system

C Contact Information

Patrick Ziajski
Phone: (647) 339 2847
Email: pziajski@myseneca.ca

Zarak Khattak
Phone: (587) 226 3196
Email: zkhattak@myseneca.ca