Garage Door Opener based on Image Processing

Design Document

Done by
Mohammed Bashmmakh

Adviser
Mohammed Almulhem

Coordinate
Ahmad Khayyat

Introduction

These days, every thing is moving toward automating. we used to to deal with every thing manually, for example we used to open the garage manually, meaning stop the car close to the door then open to it enter the car then close the door, before announcing the remote to the garage. The system approach the same idea in easy and automation way by recognising car's plate. The biggest advantage of automation is ensuring quality and consistency and not forgetting security. the project is automating the functionality of the garage systems with using a uniq sign for opening, each car has its own plate, for security and identifying.

Problem Statement

Insuring that no one expect the owner of the house enters it, by identifying him by his car's plate number, in a easy and automating way. there is some negative impact that might happen, like misreading the plate, which will result in either not allowing the owner or allowing others to enter the garage.

Project Specifications

System requirement:

- Should open the gate in a short time.
- Notify the user when recognition is succeed or failed.
- An automation for exiting the garage.
- Securely identification for the owner.
- Enough time between opening and closing safely.

Specifications:

- Stay ideal until car approaches the garage.
- Take a photo for the plate when the car is close to the garage.
- Processing the photo should take maximum of 3 seconds.
- If a plate is recognise a green lamp will turn on to notify the user.
- If a plate is not recognise a red lamp will turn on to notify the user.
- There will be sensors on the other side of the garage for auto opening
- There will be a photo for each car that has a specific plate to compare it for later attempting to help in identify fake attempts.
- The garage will stay open for five second to allow until the car fully enter.

System Design

Architecture

There are two modes, Entering Mode and Exiting Mode. The entities thats inside the micro-controller are software components and the others are hardware components.

Entering Mode

This mode is responsible of opening the gate, to allow a car to enter. When a car is approaching the garage, the system will take a picture of the car and then process it to get the plate information. Then compare it with registered plates to approve or disapprove it, and send a signal to the garage controller to open, if approved.

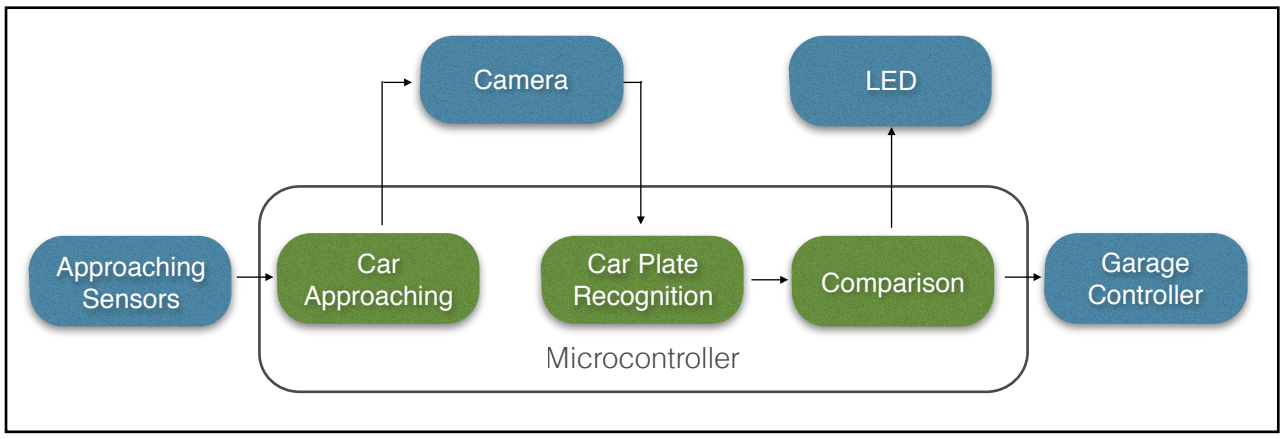

Figure 1: The architecture of system on Entering mode

Exiting Mode

This mode is responsible of opening the gate to allow the car to exit from the garage. If a car is close to the gate form the back, then the gate will open for exiting.

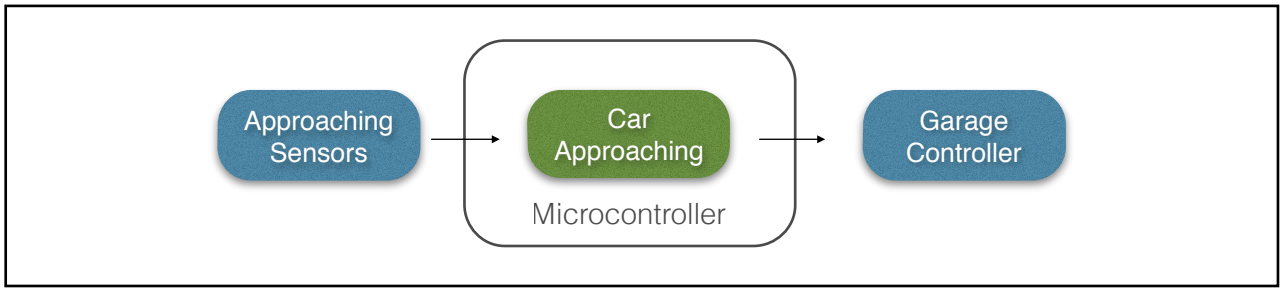

Figure 2: The architecture of system on Exiting mode

Component and Implementation

There are 4 Hardware components:

- *Approaching Sensors* : Its main job is to notify the system that a car is approaching the garage.
- *Camera* : The camera will capture an image of the plate.
- *LED* : Turn on to inform the user if a car has been recognised or not and the mode if it is in Entering or Exiting Mode.
- *Garage Controller* : The controller that controls the garage either open or close it.

There are 3 Software components:

- *Car Approaching* : To tell the system that a car is on the right place to take a picture.
- *Car Plate Recognition* : Takes an image as an input and process it to recognise the letters and numbers of the plate, and outputs a plan text.
- *Comparison* : Comparing the text, the output of Car Plate Recognition, with the registered ones.

The physical components are :

CMUcam3

It is an ARM7TDMI based fully programmable embedded computer vision sensor. The main processor is the NXP LPC2106 connected to an Omnivision CMOS camera sensor module. Custom C code can be developed for the CMUcam3 using a port of the GNU toolchain along with a set of open source libraries and example programs. This part is responsible for capturing the image and process it and sending a pain text, plate's numbers and letters. to Arduino, through serial port.

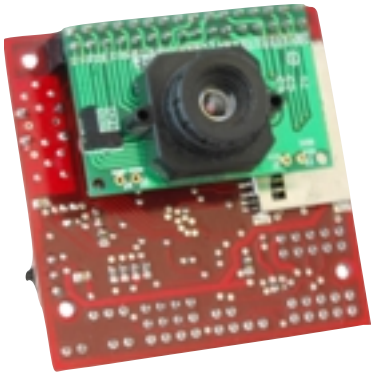

Figure 3: CMUcam3 board

Arduino UNO

It is a single-board micro-controller, intended to make building interactive objects or environments more accessible. This board is the heart of the project. It gets the information from the approaching sensors and then send a signal to CMUcam3 to capture the image and analyses it to get the plain text, then compare it with the registered plates. Then if approved, a LED will turn on and a signal will send to the garage controller.

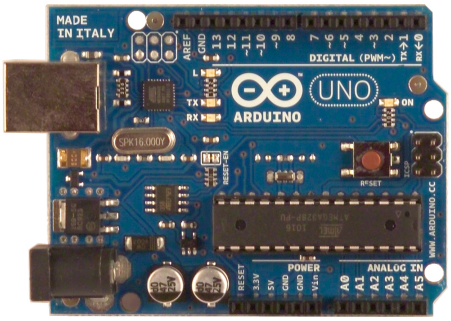

Figure 4: Arduino board

Ultrasonic Sensors

This sensors will be connected to Arduino to sense if a car is approaching.

Figure 5: Ultrasonic Sensors

LEDs

LED will be used to notify the user if the plate is recognised or not and also for identifying the mode and as the garage controller for telling the state of the garage, opening and closing .

Figure 6: LEDs

The software packages are :

OpenCV

OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision, It is free for use under the open source BSD license. The library is cross-platform. It focuses mainly on real-time image processing. it will be integrated in CMUcam3, since it is the part for processing the image. Use it to focus on the plate itself and on identifying fake attempting.

Tesseract

It is an optical character recognition engine for various operating systems. It is free software. It will also run in CMUcam3. to recognise the letters and numbers on the plate.

System Integration

Since Arduino is the heart of the project, everything will connect to it, CMUcam3 with serial port and the others, LEDs and ultrasonic sensors, through pins. CMUcam3 will have the image processing software, OpenCV and Tesseract. the first step of implantation is to install all software components on CMUcam3 to test it. Then programming both, Arduino CMUcam3, for communicate through the serial port, so we can finalise the implantation part.

Design Decisions

This system uses CMUcam3 as an image processing unit, since it has a powerful processor and API thats deals with images easily, and it is cheap and available. Arduino, because it is cheap and easy to program. other parts are also cheap and available. software packages are being used since it is an open sources softwares.

Tradeoff

the system can use either Arduino or Raspberry as microprocessor but I chose Arduino based on the table below. Arduino is small, cheaper and it has enough RAM, less power composition and it has analog inputs that will be used with the approaching sensors. For CMUcam3, I chose it since it already available to use right now.

Microcontroller	Price	Size	Ram	Min Power	Analog inputs	Experience
Arduino UNO	~29$	2.95"x2.11"	2 KB	42 mA (0.3w)	Yes	Yes
Raspberry Pi	~35$	3.37"x2.125"	256 MB	700 mA (3.5w)	N/A	Not

Table 1 :Tradeoff table

Progress

Task Schedule

Tasks	Description	Week
Collecting part	Collecting physical and software parts.	Week 8,9
Implementation on CMUcam3	installing needed software on the chip and testing it.	Week 9,10
Connecting the parts	Connecting the microcontrollers, LEDs and approaching sensors.	Week 11,12
Testing the prototype	testing the system.	Week 14
Solving other problems	Solvring other problems like fake attempting and others.	Week 15

Table 2 : Task schedule

Completed Task

- Searching for needed parts.
- Verifying the requirement.
- Study other options.
- Researches about the other approaches.