Coppi

Programming for the Internet of Things Capstone

Victoria Matute Arribas – vimarribas@gmail.com

Final project:
Specialization on Programming the Internet of Things

Supervisor:
Professor Ian Harris
Department of Computer Science

October 20, 2016

Contents

1 Introduction 2

2 Requirement specification 3

2.1 User's perspective . . . 3

2.2 Design's Perspective . . . 4

2.3 Constraints . . . 6

3 System Design 7

3.1 System Options . . . 7

3.1.1 Networked surveillance cameras . . . 7

3.1.2 Surveillance car . . . 8

3.1.3 Object recognition drone . . . 8

3.2 System Evaluation . . . 10

3.2.1 Networked surveillance cameras . . . 10

3.2.2 Surveillance car . . . 10

3.2.3 Object recognition drone . . . 10

3.3 System Choice . . . 11

4 Test procedure 12

4.1 Testing of Components . . . 12

4.1.1 Ultrasound sensor . . . 12

4.1.2 Motor(s) . . . 12

4.1.3 Wifi dongle . . . 12

4.1.4 Communication with the central surveillance center . . . 12

4.1.5 Camera . . . 13

4.1.6 Car controller . . . 13

4.2 Integration Testing . . . 14

1 Introduction

Imagine a surveillance system for a building. Typically, there are guards running around the building to make sure everything is ok. This might lead to problems in control accuracy and tiredness given by humans.

By designing a robot capable of detecting anomalies in the environment and connecting to a central surveillance service, the surveillance quality increases.

Coppi is the robot acting as the building guard. What paths should it take around the building? How will it communicate with other guard-robots? How will it detect anomalies? Thanks to all the progress on communication networks and its collaboration with different industries, Coppi will communicate all the anomalies to a central surveillance centre.

The document includes the requirement specification and design system. The requirement specification describes the behavior from a user's perspective as well as a designer's perspective.

Keywords: cloud, 5g, computer vision, ultrasonic sensor, raspberry pi,

2 Requirement specification

2.1 User's perspective

In this section, it is described how Coppi Fig. (1) works from a user perspective.

Coppi's intention is to monitor the activity in a building. Coppi has a power button, a pair of wheels and some sensors.

Coppi works in a company called FaceCatch. When most of the employees of FaceCatch go back home in the evening, Coppi will start working. Coppi has been working in FaceCatch for a long time. It knows the appearance of the offices when there are empty.

It's 20:00 o'clock. Coppi is turned on. Coppi looks around and decides walking in a given direction. As a human, it knows when it is aproaching an obstacle. When it recognises an obstacle, it looks around and decides walking in another direction. In the meantime, it studies the obstacle. If it's a person or part of a human body, it sends this information to the central surveillance centre immediately. If it finds a lot of non-expected obstacles, based on its previous days history, it sends this information also to the central surveillance centre.

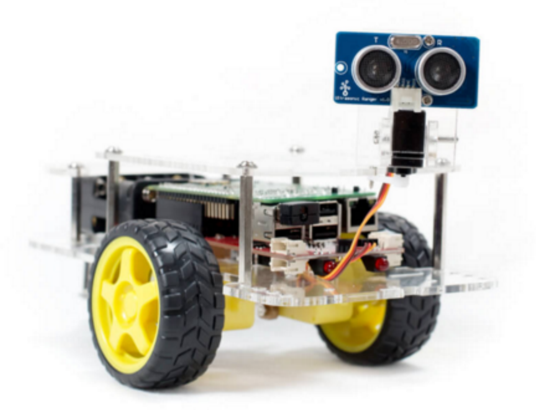

Figure 1: Coppi

2.2 Design's Perspective

In this section, it is described Coppi's design.Fig. (2)

Coppi has the following Hardware components:

- Smart car chassis
- Raspberry pi
- ultrasound sensor
- Rpi camera module
- Servo

This specialization courses will help in the design of Coppi in areas like Python, interfacing with camera module and interfacing with servo motors (PWM).

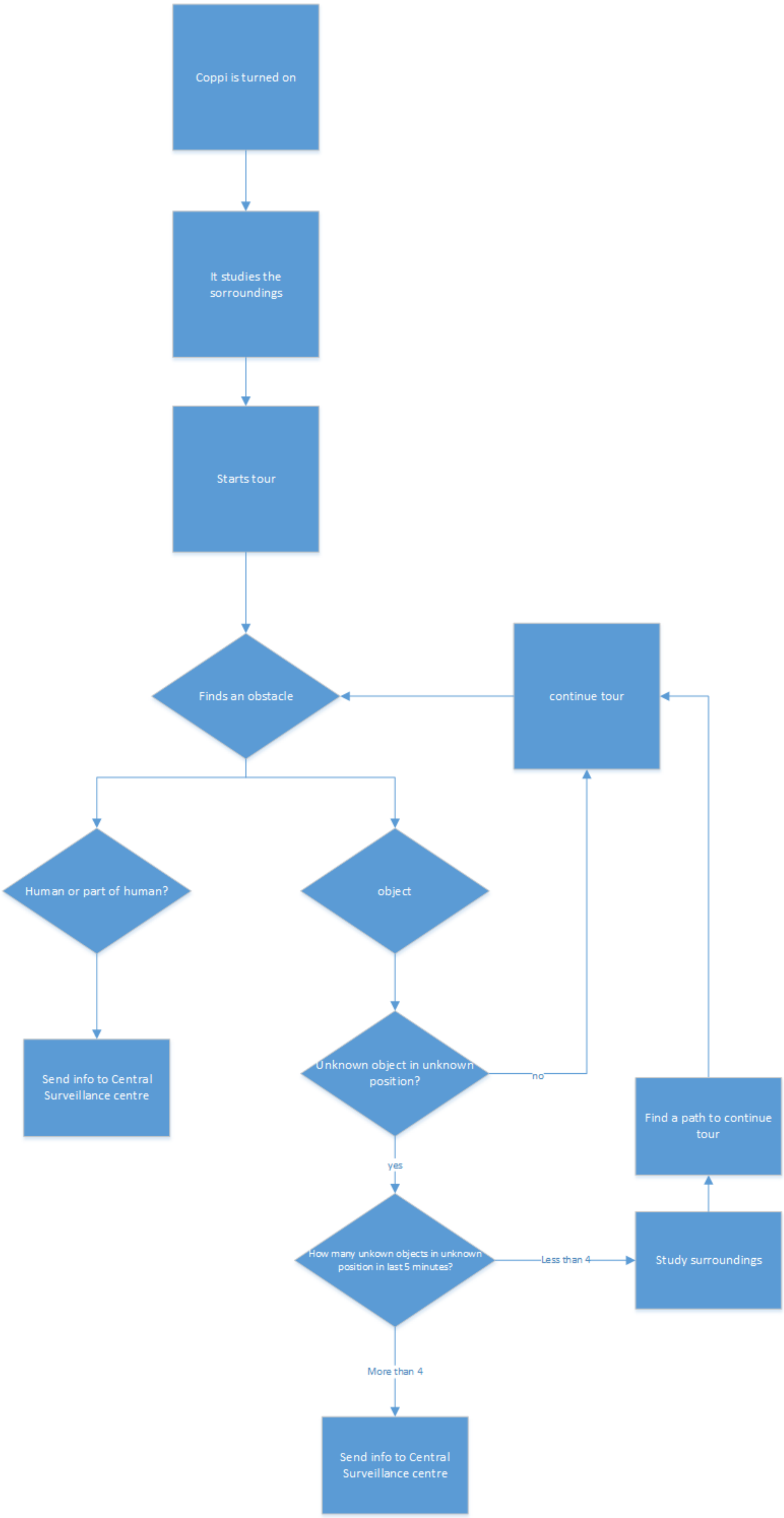

Figure 2: Coppi's Design

2.3 Constraints

In this section, it is described the constraints that the implementation of Coppi might have.

In order to build Coppi, around 110 €are needed. All details are specified in Table (2)

Component	Function	Cost (€)
Raspberry pi	Processor	40
Ultrasound sensor	Obstacle detection	7
Rpi Camera	Object recognition	26
Smart car chassis	Physical structure	27
Servo	camera and ultrasound scan	13

Table 1: Cost constraints

For the implementation of Coppi, the following steps are recommended:

1. Build up the car
2. Communication system Coppi-Central surveillance centre
3. Sensors and actuators integration in Rpi
   (a) Camera module
   (b) Servo module
   (c) Ultrasound module
   (d) Motors module
4. Implement and test the object detection and recognition software in Rpi
5. e2e Integration and verification

This would require a time frame of 2 months implementation work, 2 hours-day.

Both the communication and processing performance of Coppi should be considered. The communication performance through the central surveilannce centre should be fast thanks to 5g networks and the short messages sent. However, the post processing of video, i.e object detection and recognition, might take some minutes.

3 System Design

3.1 System Options

Coppi main functions are:

- Control over a whole room
- Object recognition
- Communication with the central surveillance centre

3.1.1 Networked surveillance cameras

Network surveillance cameras,Fig (3), consist of multiple cameras, installed in the building of the company and connected through camera controllers, i.e Raspberry pi's.

The cameras send the video to the camera controller, as depicted in the Block diagram in Fig. (4) .Each camera has a position given by GPS, known by the camera controller.The camera controller process the video, i.e object recognition. The system controller receives the objects detected and the location of each camera. The system contoller will comunicate with the Central Surveillance centre. There are several camera controllers because each rapsberry pi can handle two cameras, a USB camera and the raspberry pi module.

Figure 3: Networked cameras

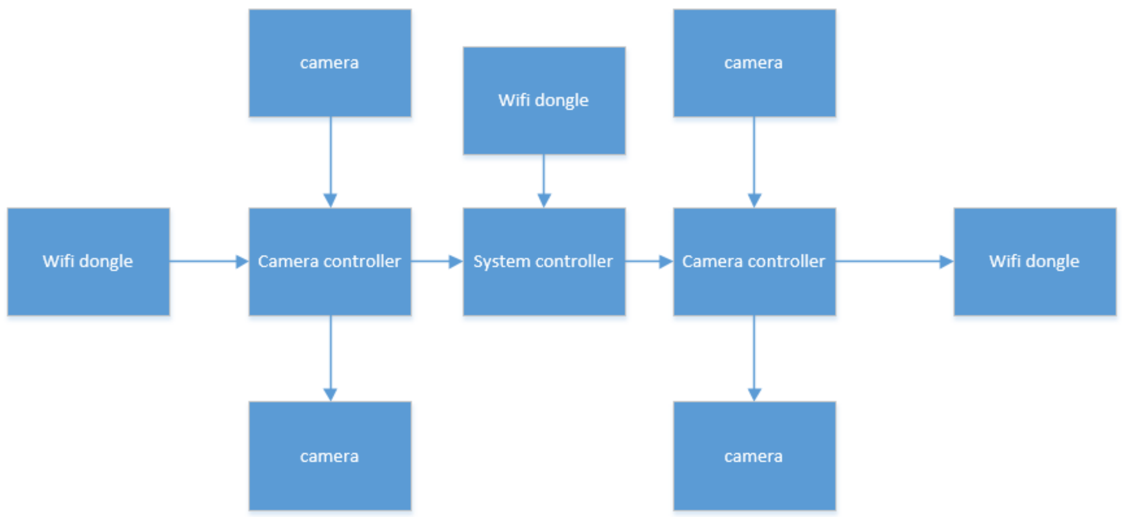

Figure 4: Networked cameras block diagram

3.1.2 Surveillance car

This is the system depicted in Fig (1). It is a car moving around the company floor and checking if there is any anomaly.

In Fig. (5), it is depicted the interfacing of each hardware component. Raspberry pi is considered as the car controller. The servo is mounted on top of the camera so that the camera can cover 360 degrees of the scene. As the Rpi will be moving, it needs some batteries.

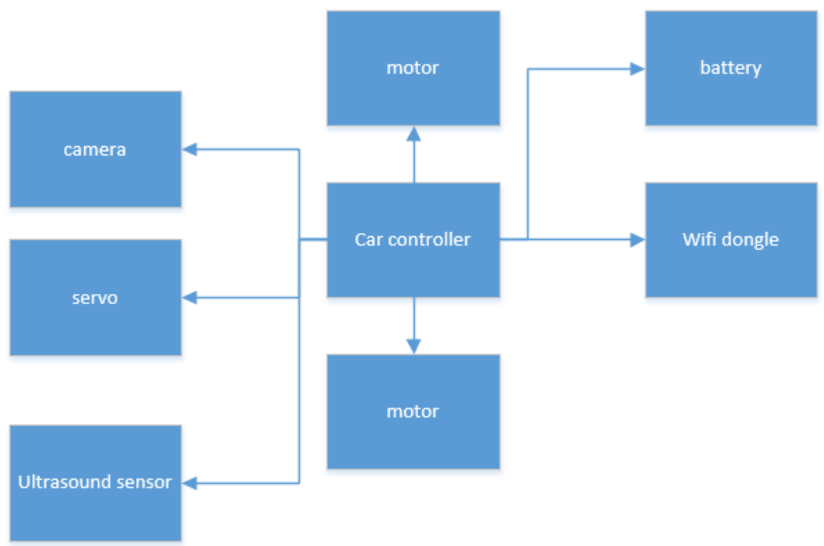

Figure 5: Surveillance car block diagram

3.1.3 Object recognition drone

The object recognition drone, Fig. (6), is the same idea as the surveillance car, in Section (3.1.2) but mounted on a drone. The advantage is that the drone covers also height, having a 3D map of the offices.

Figure 6: Drone Surveillance

In Fig. (7), the drone surveillance system is described in blocks. Raspberry pi takes the flight control, controlling 4 motors and the ultrasound and camera modules.

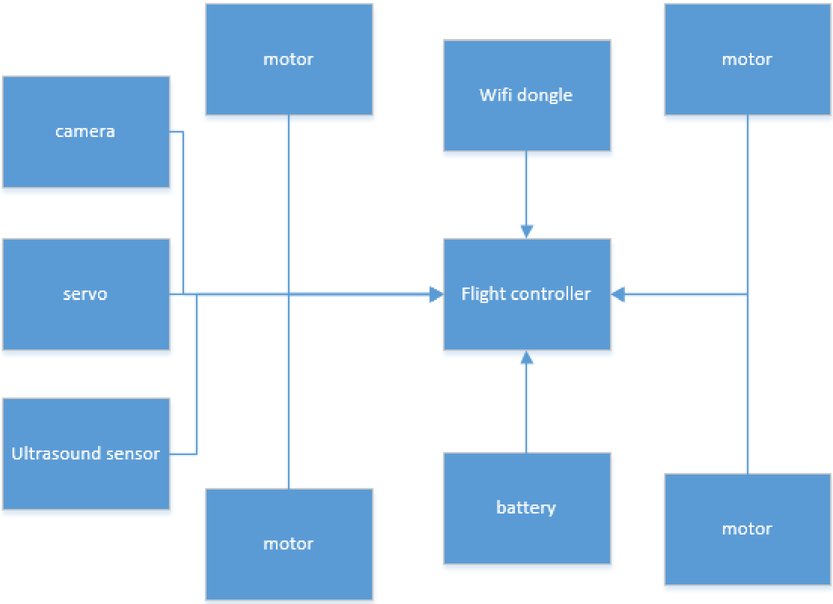

Figure 7: Networked cameras

3.2 System Evaluation

Some of the criteria to consider when evaluating the system are:

- Cost
- Size
- Measurement accuracy
- Security

  Security refers to security among this device and other human beings in the building. While the NW cameras are fixed in the wall, both the car and the drone are moving. Rules of interaction need to be established.
- Installation easy
- Portability
- operation and Maintenance
- Adaptability

  Adaptability to other buildings. While the algorithm for the car and the drone might be the same, the cabling for the networked cameras might take a lot of effort.

Design	Cost	Size	Difficulty	Measurement	Security	Installation easy	Portability	O&M	Adaptability
NW Surv cameras	370	Large	High	Low	High	Low	Low	High	Low
Surv car	100	Small	Medium	Medium	Medium	High	High	Low	High
Surv drone	250	Small	Medium	High	High	High	High	Low	High

Table 2: Evaluation criteria

3.2.1 Networked surveillance cameras

It's the more expensive and inflexible solution. The measurements accuracy is not very high since the cameras are only taking an angle.

3.2.2 Surveillance car

This solution is cheaper than the drone solution. However, it is not as accurate as the drone and it's much adaptable solution as the networked cameras.

3.2.3 Object recognition drone

This is the most flexible and accurate solution. However, there is still some security issues that need to be established in order to allow a drone fly inside a building. Human- machine interaction on air needs to be better defined.

3.3 System Choice

For learning purposes, both the Object recognition drone (3.1.3) and the Surveillance car (3.1.2) are a good match. A lot of cabling is involved in the networked surveillance camera, (3.1.1) design, which doesn't bring as intelligence as the other solutions.

Both the cameras and the drone will be pre-fabricated while the car gives the option to do it yourself, as it would be nice to fix the components, i.e Rapsberry pi, in a stable way in the car.

Due to cost, security, measurement accuracy, installation easy and portability, the Surveillance car is the best match for bringing Coppi to the real world.

4 Test procedure

4.1 Testing of Components

The components in this sytem are:

- ultrasound sensor
- motor(s)
- wifi
- Communication with the central surveillance centre
- camera

  The camera should be able to detect an obstacle and recognise if it´s a human or part of the office equipment.
- Car controller

  The car controller should differentiate between known objects and unknow objects. Given an object, the system could recognise if it´s part of the office material or if it´s new. If that object is in its position or if it has been moved.

  The car controller should also be able to drive the 4 motors.

4.1.1 Ultrasound sensor

The ultrasound sensor could be tested by driving the car into a wall. The system should recognise that there is a wall in a given distance.

4.1.2 Motor(s)

By controlling the car remotely, the motors of the car could be tested, by giving different direction and different speed.

4.1.3 Wifi dongle

Coppi should have wifi connectivity everywhere in the building. The Wifi dongle should provide connectivity. Authentication to the network and enough coverage of the network could be tested by driving the car all around the building.

4.1.4 Communication with the central surveillance center

An intentional message could be send to the central surveillance center.

4.1.5 Camera

By locating different people and objects in front of the camera, it should recognize their categories. The camera should also be able to recognise parts of humans. It would be a good test to position some parts of the body, i.e from mannequins, in front of the camera.

4.1.6 Car controller

The car controller should be able to drive the car by itself, by knowing the map of the building. In order to test this functionality, the car with the car controller should smoothly drive by itself all around the building.

The car controller should recognize familiar and unfamiliar objects, objects in its position and objects out of its position. A possible test would be include two scenarios: locating different objects in the floor (objects that normally are in the desk) and a tidy scenario, where the objects in the floor are just a chair, table, carpet...

4.2 Integration Testing

Coppi integration testing is incrementally. The following order is recommended:

1. motor(s)

   The car should be able to move, left, right, forward and backwards.

2. motors + ultrasound sensor

   The car should be able to drive without hitting any wall, being controlled remotely

3. motors + ultrasound sensor + wifi

   The car should be able to connect to the wifi from the building

4. motors + ultrasound sensor + wifi + car controller

   The car should be able to drive autonomously around the building.

   If taken to another path which it doesn´t know, it should be able to come back to known area.

5. motors + ultrasound sensor + wifi + car controller + camera

   The car should be able to recognise objects in its path around the building.

   Testing different density of objects at different speed would allow to test the accuracy of the system.

6. motors + ultrasound sensor+ wifi+ car controller + camera + communication with the central surveillance centre

   To test the whole system, Coppi should identify an object which is not in its usual position. Immediately, it should communicate with the central surveillance centre.

References

[1] *cloud*, http://www.dexterindustries.com/gopigo/

[2] *cloud*, http://googlecloudplatform.blogspot.de/2015/12/Google-Cloud-Vision-API-changes-the-way-applica html

[3] *HW*, http://www.dexterindustries.com/shop/gopigo-starter-kit-2/

[4] *Design options*, http://news.mit.edu/2015/object-recognition-robots-0724

[5] *Surveillance systems*, https://www.samsung-security.com/SAMSUNG/upload/Product_Specifications/Samsung_SGR_Series.pdf