REAL-TIME VISION-BASED FISH MONITORING SYSTEM USING RASPBERRY PI

BY

WAYNE FONG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2022

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Real-time vision-based fish monitoring system using Raspberry Pi____

_____

_____

Academic Session: _____2021/2022_____

I _____WAYNE FONG_____

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

_____

(Author's signature)

_____

(Supervisor's signature)

Address:
New Town Ship. Lot 32_____
Restaurent Menawan._____
P.o.box 169. 89907 Tenom, Sabah

Date: _____19/4/2022_____

Gan Ming Lee

_____

Supervisor's name

Date: 21/4/2022

FACULTY/INSTITUTE* OF Information and Communication Technology

UNIVERSITI TUNKU ABDUL RAHMAN

Date: _____19/4/2022_____

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that _____***Wayne Fong***_____ (ID No: ***18ACB02229***) has completed this final year project/ dissertation/ thesis* entitled "_____*Real-time vision-based fish monitoring system using Raspberry Pi*_____" under the supervision of _____Ts Dr Gan Ming Lee_____ (Supervisor) from the Department of Department of Computer and Communication Technology, Faculty/Institute* of Information and Communication Technology.

I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

_____

Wayne Fong

DECLARATION OF ORIGINALITY

I declare that this report entitled "**REAL-TIME VISION-BASED FISH MONITORING SYSTEM USING RASPBERRY PI**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : 

Name : Wayne Fong

Date : 19/4/2022

ACKNOWLEDGEMENTS

I would like to express thanks and appreciation to my supervisor, Ts Dr Gan Ming Lee and my moderator, Ms Tan Lyk Yin who have given me a golden opportunity to involve in the embedded system field study. Besides that, they have given me a lot of guidance in order to complete this project. When I was facing problems in this project, the advice from them always assists me in overcoming the problems. Again, a million thanks to my supervisor and moderator.

Other than that, I would like to thank my project teammate, Terry Teh who has provided a lot of assistance to me when completing this project. Although both of us are having different project and task scope, he is still willing to support me when I faced difficulties in developing this project.

ABSTRACT

In this paper, the studies towards embedded system, machine learning, and the foundation of electronic circuit have been made. In the modern days, there are many aquaculture farms being developed and maintained to produce more fishes for production. To increase and maintain the performance of the farm, many have deployed various approaches to monitor the fishes in the farm. However, the monitoring process sometimes could be dangerous if proper protocol is not followed. Besides, with only raw human power, the effectiveness of the monitoring process could be reduced greatly. To enhance the productivity, the study of the machine learning would be used for developing a fish detection system. This system would be used as a program to provide users a visual monitoring services to detect the fish entity from an image. At the end of the development, the performance of the detection model in terms of detection accuracy is over 82% mAP. To run the detection system, the system is being embedded into an embedded computer called Raspberry Pi. Furthermore, it is still possible to improve the overall system by adding a few more features. In this case, a simple electronic circuit with the controls from an Arduino microcontroller has been designed to simulate on how the overall system could be improved by implementing a temperature and light level monitoring system. With such implementation, this allows user to monitor not only the fishes activity, but also the temperature of the environment to prevent the fishes from various illness and the light level monitoring system could be used as an indication if more light is required for better visual monitoring process. To allow user to use the system from a system, the system has been configured with VNC services for remote access.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
CPU	Central Processing Unit
GPIO	General Purpose Input Output
IOT	Internet of Things
IP	Internet Protocol
RAM	Random Access Memory
CNN	Convolutional Neural Network
OS	Operating System
ANN	Artificial Nueral Network
GMM	Gaussian Mixture Models
YOLO	You Only Look Once
RPN	Region Proposal Network
CUDA	Compute Unified Device Architecture
API	Application Programming Interface
ReLU	Rectified Linear Unit
mAP	Mean Average Precision
SBC	Single Board Computer
FYP	Final Year Project
FPS	Frame Per Second
SSH	Secure Shell

CHAPTER 1

Introduction

Nowadays, people are living in a happy and confortable environment as technology grows. Besides, with the improvement of technology from time to time, it has become one of the most important elements for humans no matter in which specific area such as education, development and even survival of human. Furthermore, with the current technology, it has also contributed in the increasing number of population across the entire world. According to studies, it has been recorded that there were over 6 billions people living on earth in year 1999 and the number of population has been increased to over 7 billions in year 2020. Moreover, it is also predicted that there will be over 10 billions population living on earth in year 2057. Based on the situation as mentioned earlier, it is obvious that food supply would be a very important aspect to be concerned in order to preserve the survival of people around the world. Because of that, many countries across the world have been starting to make investment in food production sector in order to keep themselves to have enough food supply for the people in the country. Speaking of which, food production sector could be referring to farming such as crop farming, chicken farming, and even fish farming or so called the aquaculture farming. In addition, due to the great improvement in technologies, some of the farming sectors that used to be difficult to maintain and monitor especially in aquaculture farming has become a lot better with the help of technologies in monitoring process. Furthermore, aquaculture farm is considered as one of the most important element in many countries not only in food production to maintain the food supply, but it is also very important in captive breeding which involves the process of breeding the animals in a controlled environment. With such, thousands of aquatic species still could be preserved and used in food production while preventing them from extinction.

1.1 Problem Statement and Motivation

As what has been discussed earlier, aquaculture farm is a very important element in the industry to breed and cultivate more aquatic cultures not only for fulfilling the high demand in food supply, but it is also the key farm that could preserve many species of aquatic creature from extinction due to the high demands of food supply caused by the increasing of population across the world. Besides, even in the current era with great technologies being developed,

there are still countries that are having the issue of food shortage due to plenty of direct and indirect reasons. For example, according to the World Food Program USA (UN WFP) [1], Yemen is one of the countries that is facing the issue of hunger crisis and there were over 16 millions of people do not have enough food just to simply fulfill food hunger everyday and there were more than 2 million children severely malnourished. With such situation, most countries do aware that food supply is very important in order to preserve the survival of their people in the country. Furthermore, as the scientific researches grow, there are ways being proposed in order to have greater throughput in terms of the rate of aquaculture generation. However, even though there are many ways being proposed that could help in better generation of farming, the monitoring process became a serious problem in order to keep the farm continues to run in the long run. In addition, the monitoring process could be referring to the performance of the process in terms of effectiveness, and most importantly, the safety of the process. Furthermore, according to some existing aquaculture monitoring systems, most of the farms perform the monitoring process by monitoring the water temperature, pH level of the water, and the level of dissolved oxygen (DO) in the water by using specific sensors. Despite that, it is not deniable that the monitoring process as mentioned earlier are only able to monitor the quality of the environment in the water. In other words, this does not solve the problem where the activity of the fishes could be monitored properly. Another alternative way used by most farms to monitor the activity of the aquatic creatures is by assigning several employees to monitor and watch the farm nearby the farm location. However, such action could place the employees at safety risk. For example, if the aquaculture farm is a farm that cultivates electric eels, the employees that monitor around the farm may be falling into the pool and might be killed by the eels since an adult eel could produce up to 600 volts of electrical energy which is enough to kill an individual. Therefore, in this project, a system that provides visual monitoring services would be studied and developed in order to contribute a safer and a more effective method to perform aquaculture farm monitoring process to societies and communities.

1.2 Objectives

In the previous section, it is discussed that fish monitoring process could be less effective with only the sensors to monitor the condition of the water in the aquaculture farm without properly monitor the activities of the fishes at the farm location. Besides, monitoring the fishes at the farm location may bring unwanted situation that could cost someone's life. In order to overcome the issue as mentioned earlier, a system that could provide visual monitoring services

by utilizing the technology of artificial intelligent (AI) would be studied and developed in this project. Furthermore, the main idea of the proposed system would be very similar to the proposed system developed in Final Year Project (FYP) 1. However, due to the limited time and experience, there are still rooms available to be improved based on the solution proposed in FYP 1 in order to enhance the overall system to become even more useful not only to people that are interested in studying embedded system, but also the people that are interested in deploying the system in real life industry for monitoring their aquaculture farm. Meanwhile, it is also noticed that by only having a system that provides visual monitoring services to monitor the activity of the fishes in an aquaculture farm is not truly enough in the industry. For that reason, this project would be carried out by studying more on not only the improvement of AI, embedded system, but also a new subsystem that involves electronic circuit design which shall be discussed further in the following chapters. Therefore, in order to successfully complete this project, project objectives should be well defined to act as a stopping point of this project. The following is the objectives that would be achieved in this project.

1. To develop a fish monitoring system that provides visual monitoring services.
2. To develop a fish monitoring system that allow users to safely monitor fishes visually without staying at the farm location.
3. To develop a FYP 1 based monitoring system with better enhancement in terms of AI detection accuracy and functionality.

1.3 Project Scope

In this section, the scope of the project would be discussed further. Based on the discussion made in the previous section, it is mentioned that an enhanced fish detection in terms of detection accuracy would be studied and developed in this project. Besides, it is also mentioned that one of the objectives in this project is to develop a fish monitoring system that allow users to safely monitor the fishes without staying at the farm location. Based on the objective, this statement would be a very important element in this project to enhance the overall system usage in terms of availability and further information would be discuss soon. Not only that, a new subsystem would also be added in this project to enhance the overall functionality of the whole system. To understand the statements as mentioned earlier, the following is the project scope with explanation included.

1. To develop a fish monitoring system that is capable of providing necessary visual monitoring services with a single board computer named Raspberry Pi that uses a camera module to capture images of fishes.
2. To develop a fish monitoring system that is capable to detect the existence of fish entities captured in an image frame.
3. To develop a fish monitoring system with the performance of detection module with at least 70% of mAP (mean average precision), and recall metrics of at least 50%.
4. To develop a fish monitoring system that is capable of measuring the environment temparature with a simple temperature sensor.
5. To develop a fish monitoring system that could detects surrounding light level and trigger a LED for indicating if more light is required for better monitoring process.
6. To develop and design a simple electronic circuit to emulate and support the functionality as mentioned in scope 4 and scope 5 with an Arduino microcontroller.
7. To develop a fish monitoring system that allow users to use the system within a local area network by utilizing the functionality of available remote access built in the Raspberry Pi.

Based on the project scope listed above, there are 7 scopes to be achieved in this project. Besides, based on the first project scope, the whole system would be deployed and hosted with a single board computer named Raspberry Pi with a camera module attached to it. With the camera setup, a fish detection module could be developed and embedded into the Raspberry Pi to work with the camera module as mentioned earlier. Besides, this project will also involves further research in developing a detection module with at least 70% of mAP and 50% of recall metrics in order to allow the system to properly detect the specific fish entity from an image frame. Moreover, during the research process, it is noticed that visual services may not necessary be enough in monitoring the fishes in aquaculture farms. Because of that, it is planned to develop a simple temperature monitoring subsystem since most fishes do not have the same requirement for environment temperature to live. Also, it is worth mentioning that a light sensor would also be implemented in this project because different lighting environment may affect the performance of the fish detection model in terms of accuracy. For that reason, a simple LDR light sensor would be used in the system to act as the component to sense the light level in the environment and trigger specific LED whenever the environment gets darker to indicate that more light is required for better fish detection process. Also, to implement the newly added subsystem such as the temperature and light sensors, the main module that would

be used to control those components would be a famous microcontroller named Arduino and the further reason would be discussed further soon. Last but not least, it is mentioned in the project objective that the fish monitoring system would be developed to be usable without the need of users to reach the farm location. Therefore, this problem could be solved by enabling and setting up the remote access service in the Raspberry Pi to allow users to use the system within a local area network. In short, the overall implementation of this project is to enhance the performance and the functionality of the proposed system in FYP 1.

1.4 Contributions

In this project, the whole deliverable that would be generated is a fish monitoring system that involves with the deployment of Raspberry Pi, a single board computer with the development of AI detection for detecting fish entities from an image captured by the camera module installed in the system. Besides, the whole system also consists of a few modules such as temperature and light sensor that would act as the supportive module to enhance the overall performance and the functionality of the proposed system. In orther words, the monitoring system works by performing the fish detection with the help of the detection module deployed in the Raspberry Pi along with the camera module installed in the system. Besides, due to some existing weaknesses found in the proposed system in FYP 1, some additional modules such as a temperature and a light sensor have been included in the system in order to improve the overall system. Furthermore, to support and control the newly added modules, a famous microcontroller has been deployed in the monitoring system called Arduino. In short, this project involves designing and developing electronic circuits with proper control configuration with specific programming tool.

Moreover, with the development of technologies and the improvement of scientific researches in farming sector, many farms are able to generate and cultivate more deliverable especially in aquaculture farm for various reasons. To maintain the farm in long run, many people have made lots of investment in deploying various methods to watch over their farm to keep their farm working propely while not degrading the throughput of the system. Based on some existing aquaculture farm, most of the common used technologies are sensors such as temperature sensor, pH sensor, and even disolved oxygen oxygen to make sure the fishes in the farm are in a comfortable environment. However, there are still ways to improve the monitoring process by utlizing the existing technologies such as a microcontroller and a single board computer. In addition, the improvement of the process does not only enhance the

effectiveness of the monitoring process, but also the safety of the whole monitoring process. Therefore, based on the proposed system in this project, it could be a great contribution to people especially IT students, IT researchers and developers that are interested in the study of image processing, computers, fundamental of networking, and fundamental of electronic.

1.5 Report Organization

In this report, it is organised into 7 chapters: Chapter 1 Introduction, Chapter 2 Literature Review, Chapter 3 System Model, Chapter 4 System Design, Chapter 5 System Simulation, Chapter 6 System Evaluation And Discussion, Chapter 7 Conclusion And Recommendation. In Chapter 1, the introduction of aquaculture farm and the project including the problem statement, motivation, objectives, scopes, contributions have been discussed and explained. In Chapter 2, various technologies and existing systems that is similar to the project are studied and discussed to develop a solution for this project. In Chapter 3, the idea of how the proposed system functions is discussed including the system use case. In Chapter 4, the implementation and the development of the proposed system is being discussed and documented. In Chapter 5, the system simulation is performed to show how the proposed system could be prepared and used for demonstration purposes. In Chapter 6, several evaluation have been made towards the proposed solution and the challenges faced during the system development has also been discussed. Besides, this chapter has also perform evaluation towards the achievement of the project objectives in order to identify if project objectives are achieved. Last but not least, the conclusion and the recommendation for this project have been discussed in the final chapter which is Chapter 7.

CHAPTER 2

Literature Review

Based on the discussion made in the previous chapter, it is discussed that a fish monitoring system with specific requirements would be studied and developed. Besides, while researching and developing the system, there are literatures were being reviewed and became one of the motivation to further proceed the project. In the same time, it is also discovered that there are many options available to develop the whole system with various method to improve the system. Therefore, decisions on "what to do" and "how to do" have been made with the optimal solution based on the scenarios as discussed in the previous chapter. The following will discuss more about the review of the technologies used and the existing system studied in this project.

2.1 Review of the Technologies

2.1.1 Laptop

In this project, it involves the development of AI detection module which is the main software that would be used in the proposed system to perform and provide most of the visual monitoring services to users. In this case, a laptop or a computer is indeed required to develop the detection module. Besides, during the development process, it involves with the training session for the detection module in order to learn the necessary details of the object that it should be detecting during system deployment. For that, a laptop with strong computational power is highly recommended to shorten the training process. Therefore, the following table shows the details of the laptop used in this project carry out the development process.

Description	Specification
Model	Asus A556U
Processor	Intel(R) Core(TM) i7-6500U CPU @ 2.50GHz - 2.59 GHz
Operating System	Windows 10, 64-bit
Graphics Card	NVIDIA GeForce 940M
Memory	4GB DDR3 RAM (2 units)
Storage	1TB SATA HDD

Table 2-1-1-1 Specification of Laptop

2.1.2 Single Board Computer (SBC)

As the title of this project suggested, it involves with the deployment of Raspberry Pi, a single board computer to complete the entire proposed system. Besides, before going into the details of the information regarding Raspberry Pi, it is important to understand the fundamental of single board computer or SBC. First of all, according to Bae Systems [2], a SBC refers to a single circuit board that contains important built in components such as a microprocessor and memory. In other words, a SBC is a fully functional computer that has all the necessary components built in in a small circuit board. Besides, the size of SBCs could be vary from each other according to the design of the boards from various companies. Furthermore, according to Long [3], there are many types of SBC with different hardware specification available in the market such as Odroid N2. Because of that, it is important to wisely choose the most suitable SBC to complete the project. Also, based on the suggestion made by Long [3], there are a few different models of SBC which are made and produced from the same company called Raspberry Pi. Besides, Raspberry Pi organization has designed and offered many types of SBC that would fit its customers' requirements based on the customer's need. Therefore, with sufficient research on the requirement of the SBC, the model of the SBC would be used to complete the proposed system is the Raspberry Pi 3B+. The following table shows the specification of Raspberry Pi 3B+.

Description	Specification
Processor	Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4GHz
GPU	Broadcom Videocore-IV
Memory	1GB LPDDR2 SDRAM
Networking	Gigabit Ethernet (via USB channel), 2.4GHz and 5GHz 802.11b/g/n/ac Wi-Fi
Bluetooth	Bluetooth 4.2, Bluetooth Low Energy (BLE)
Storage	Micro-SD
GPIO	40-pin GPIO header, populated
Ports	HDMI, 3.5mm analogue audio-video jack, 4x USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)
Dimension	82mm x 56mm x 19.5mm, 50g

Table 2-1-2-1 Specification of Raspberry Pi 3B+

Source : (https://magpi.raspberrypi.com/articles/raspberry-pi-3bplus-specs-benchmarks)

Based on Table 2-1-2-1, the Raspberry Pi 3B+ is powered power an ARMv8 processor that is capable of running at the clock of 1.4GHz. Besides, the SBC is also capable of storing 1GB of running processes with its LPDDR2 SDRAM. Furthermore, one of the worth mentioning components that contains in this SBC is that it has the capability to support the 2.4GHz and 5GHz network via ethernet or Wi-Fi connection. This feature is one of the important features that has improved the functionality of the proposed system and this would be discussed in the further chapters. Also, the size and the weight of the SBC are also the main reason that this SBC has been chosen to implement the proposed system. Based on the dimension, the length of the SBC is only around 8cm, height of nearly 2cm, width of around 5cm and only weighted 50g. With such size and weight, it shows that this SBC is very light and small which enhance the mobility of the SBC. Not only that, the SBC itself does not require high power supply in order to power up the whole SBC to work normally. Based on the discussion in the Raspberry Pi forum [4], it is suggested that a Raspberry Pi 3B+ model SBC only requires a power supply of 5V and 2.5A. Overall, the total power required based on the requirement as mentioned earlier is only around 5V * 2.5A = 12.5W. Compared to a modern desktop, usually it will require 650W to 850W of power supply which is over 54 times higher compared to a Raspberry Pi SBC [5]. Last but not least, a SBC is usually cost lesser compared to a modern desktop or computer. In this case, a Raspberry Pi 3B+ only cost around $35 which is a great opportunity for students to study further about embedded computers without the need to spend too much money [6]. In short, Raspberry Pi SBC is a low cost, small, portable, and power efficient computer that could be used for various computing task.

2.1.3 Raspberry Pi OS (Buster) 32-bit

As discussed from the section above, Raspberry Pi is a single board computer that could work and perform most basic operations like modern computers could with the exception that it is much more power efficient and compact. Because of that, a Raspberry Pi SBC would not be a complete computer without the operating system (OS) itself since the OS is the main component that is responsible to manage and allocate the resources available to run the processes created. Besides, based on the studies on the SBC, most of the Raspberry Pi SBC supports many types of OS especially for Linux OS. According to studies, many have recommended to use and deploy Linux OS in Raspberry Pi SBCs because of the how well

Linux OS manage the resources. For example, according to Linus Tech Tip [7], he has mentioned that one of the advantages of using Linux is that Linux could perform great without needing a powerful computer due to its lightweight design. Because of that, Linux is also a great OS to be used and deployed in many computer hardwares especially in a low end computers such as Raspberry Pi SBCs which uses ARM processors with lower computing power compared to modern processors. Furthermore, according to the Raspberry Pi foundation [8], the developers have developed various versions of Debian based Linux distribution specifically for all of the available Raspberry Pi SBC in order to suit the users' need. In general, the name of the OS developed by the Raspberry Pi foundation is called the Raspberry Pi OS. Based on the most recent udpate, the following is the list of available version of Raspberry Pi OS.

Index	Name	System	Debian Version
1.	Raspberry Pi OS (Legacy) with desktop	32-bit	10 (Buster)
2.	Raspberry Pi OS Lite (Legacy)	32-bit	10 (Buster)
3.	Raspberry Pi OS with desktop	64-bit	11 (Bullseye)
4.	Raspberry Pi OS Lite	64-bit	11 (Bulleyes)

Table 2-1-3-1 List of Raspberry Pi OS

Source : (https://www.raspberrypi.com/software/operating-systems/#raspberry-pi-os-32-bit)

Based on the table above, there are 4 types of different version of Raspberry Pi OS that is available for Raspberry Pi SBCs. In addition, every version of the OS are different from each other. According to the Raspberry Pi foundation, the OS named with the term "Lite" after the name of the OS refers to the OS without the desktop environment hence, the OS would be lighter compared to the OS with desktop environment. Besides, the Raspberry Pi OS could also be categorized in 2 different system type and there are the 32-bit and 64-bit systems. Both systems are developed with different version of the Debian system. In this case, 32-bit system is designed with Debian 10 system and the 64-bit system is designed with Debian 11 system. Furthermore, both 32-bit and 64-bit OS do perform differently compared to each other. For example, according to Jeff Geerling [9], 64-bit system performs faster compared to 32-bit system due to the design of the A64 instruction where it has more registers that could help in speeding up the processing time. The A64 is available in newer processor architecture such as the processor built in the Raspberry Pi 3B+. However, he also mentioned that 64-bit system

consumes memory resources compared to the 32-bit system [9]. In short, this is considered as a drawback for deploying the 64-bit system into a resource limited hardwares like the Raspberry Pi 3B+ which only has 1GB of memory. Therefore, the Raspberry Pi OS (Legacy) with desktop would be the main OS to be used in this project.

2.1.4 Microcontroller

In the 20th century, most of the tasks were being managed manually due to the poor development of technology during that time being. As time flies, technologies have become more useful and reliable with proper studies and development. With the development of technologies, most of the daily tasks used to be managed by raw man power are now being managed with cloud technology automatically and this has improved the productivity of the industries regradless in production throughput or security. Because of that, it is obvious that the knowledge related to IT and electronic should be taught to people. In some countries, some of the schools offer courses that teach electronic components and IT by letting the students to implement simple programs using a computer to control a microcontroller that is responsible to control the components in an electronic circuit. As simple as it sounds, a microcontroller integrated circuit (IC) device mostly used for controlling electronic system [10]. Normally, a person could assign instructions to a microcontroller and the instructions would be controlled by the built in microprocessor unit (MPU) on the board. During the studies, one of the most popular microcontroller discovered is called the Arduino. Based on the documentation provided by the Arduino foundation [11], Arduino is suitable and popular among middle school students, high school students and even the university students for studies and development. As the recommendation suggests, the microcontroller that will be used in this project is the Arduino Uno since it has most of the basic functionality for this project and the people that would like to study in IoT or cloud technology. The following table and diagram describes the specification and details about the pinout built in the device.

Description	Specification
Microcontroller	ATmega328P
Operating Voltage	5V
Recommended Input Voltage	7-12V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6

Analog Input Pins	6
Size	68.6mm x 53.4mm, 25g

Table 2-1-4-1 Spefication of Arduino Uno Board

Source : (https://store-usa.arduino.cc/products/arduino-uno-rev3/)

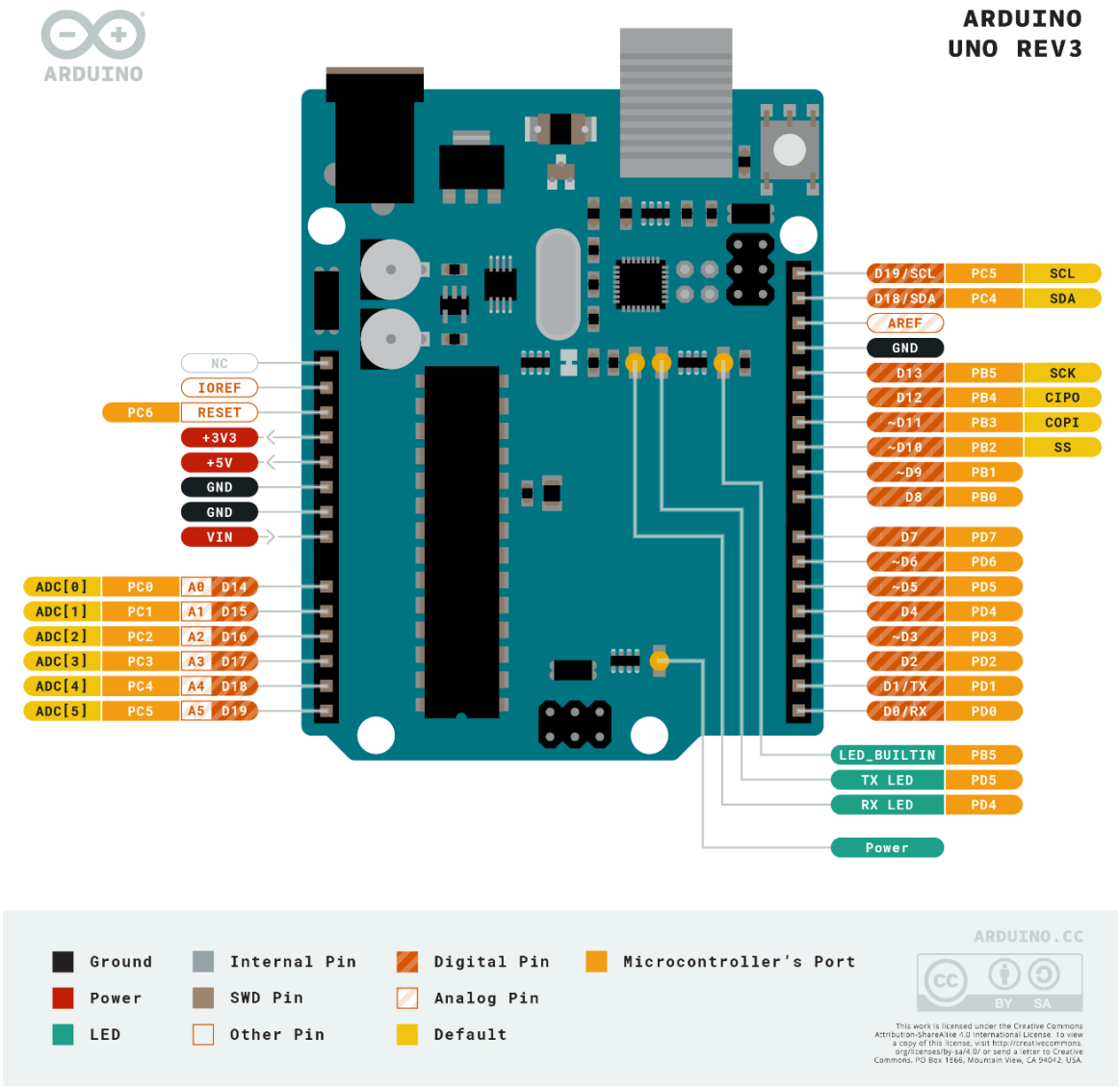


Figure 2-1-4-1 Details of Arduino Uno Pinout

Source : (https://store-usa.arduino.cc/products/arduino-uno-rev3/)

Based on the Table 2-1-4-1, there are a few reasons why the Arduino Uno is used in the proposed system. First of all, similar to most of the Raspberry Pi SBC, Arduino Uno does not require high power supply in order to fully operate. In fact, the board itself is power efficient enough to be powered with only using a 9V baterry as the operating voltage required for the board is only 5V. Besides, there are a lot of well made documentation for the Arduino Uno

provided by the Arduino foundation in using their products. This is helpful especially for those inexperienced people to learn more not only about the correct usage of the hardware, but also the safe way to use the hardwares for preventing the unwanted occasion from happening. Not only that, due to the compact design of the board, using the Arduino technology is certainly a great element to be deployed in the proposed system to enhance the overall functionality and performance without affecting the mobility of the proposed system.

2.1.5 Programming Language

In this project, programming would be the main activity to be performed in order to development the proposed system. For that reason, it is important to choose the programming language wisely according to the development process involves in this project. According to the studies made in this project, there will be 2 programming languages would be used to implement the proposed system and the programming languages are Arduino code and Python. First of all, Arduino code is programming language which is being developed based on C++ programming language with the additional special methods and functions [12]. This programming language is also the main programming language that would be used for the Arduino board in order to provide the necessary instruction for the board to perform the necessary task for controlling the electronic components which would be discussed in the further chapter. To use the programming language, the Arduino IDE could be downloaded and installed easily by following the documentation available from the Arduino foundation. Once the installation is completed, the development can begin by creating a sketch file which contain the Arduino code written by developers. After that, the sketch file is then being processed and compiled to the machine language. In short, the Arduino code is the main programming language that would be used to program the Arduino board according to the project requirement. Next, Python programming language is one of the most popular languages among others due to the syntax simplicity with wider supports from the community. Furthermore, due to the simplicity of the syntax in Python, Python has become one of the most used programming languages for developing machine learning and deep learning projects [13]. Besides, to develop the deep learning module for this project, there are a few types of famous libraries that could be used to build a deep learning module and the libraries are Tensorflow and PyTorch. Both libraries provides many useful methods and functions to build a deep learning module based on the requirements of users. For example, a user may build a deep learning module for object detection or image classification. In short, Python programming language is simpler compared

to many other programming languages and it has greater supports since it is an open-source programming language which definitely could improve the development experience and time [13].

2.1.6 Convolutional Neural Network (CNN)

Based on the discussion earlier, it is mentioned that this project involves with the development of deep learning module in order to implement the detection system for detecting fish entities from an image captured by the camera. For that reason, it is important to study the basic concept of machine learning in order to develop the deep learning module in this project. First of all, according to Brownlee [14], deep learning is a subfield of machine learning that involves the algorithms that are designed based on the function that a brain has and this technology is called the artificial neural network (ANN). Besides, ANN is a system that is developed for simulating the behaviour of a human's brain in analyzing and processing information. Because of that, ANN technology has been widely used in learning the data provided by the developers mostly for problem solving purposes. The advantage of using ANN system for problem solving is that ANN system tends to perform better in problem solving process compared to human especially when there are higher difficulty level of problems that might require beyond human's capability to solve. Other than ANN, there many different types of deep learning algorithm that serve different purposes compared to each other. In this project, the Convolutional Neural Network (CNN) would be one of the deep learning algorithms that best suits the requirement of this project since CNN is one of the deep learning algorithms that is widely used in computer vision and image analysis field [15]. Furthermore, CNN is well known in the development of computer vision and image analysis because the CNN is a class that consists Deep Neural Networks that is responsible for performing the recognition and classification of particular features from images. To understand further about CNN, below shows a diagram that describe the basic of a CNN architecture model.

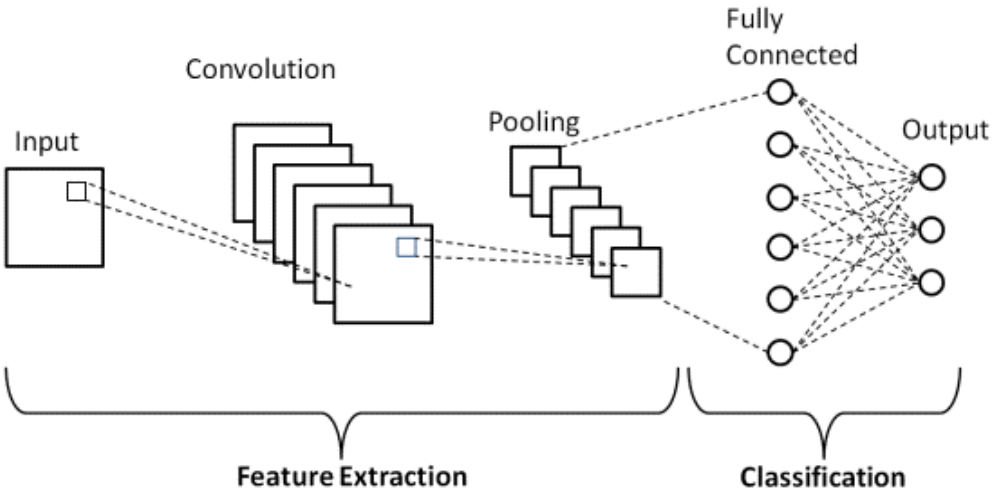


Figure 2-1-6-1 Basic CNN Architecture

Source : (https://www.upgrad.com/blog/basic-cnn-architecture/)

Based on the figure above, the architecture consists of 2 main components which are the feature extraction and the classification component. According to Gurucharan [15], when input images are fed into the CNN model, the images would then be entering the first layer of the CNN model which is the convolutional layer. In this layer, convolutional layer is responsible for extracting various features such as the edges, color from the input images fed into the system [16]. After that, the output generated by the convolutional layer, also known as the feature map would then be fed into the pooling layer which is responsible to decrease the size of the convolved feature map obtain from the previous process. In fact, this process is very important in most CNN architecture model because the main reason that pooling layer exists is that it could reduce the computational workload when the model is being deployed and trained in a computer. Overall, that would be the processes that would involve in the feature extraction section. Again, according to Gurucharan [15], the fully connected layer consists of the weight and the biases with the neurons in the architecture. Not only that, this layer is also responsible for connecting the neurons between 2 different layers as shown in Figure 2-1-6-1 in the classification section. Based on the Figure 2-1-6-1, the fully connected layer is connecting the nuerons directly to the output layer. Speaking of which, fully connected layer is also the layer that is responsible for the classification process and after completing the classification process, it would then store the output in the output layer. In summary, CNN models perform the learning process with proper training configuration in the feature extraction section and starts to perform the classification process in classification section.

2.1.7 Summary of the Technologies Review

In summary, there are total of 6 technologies being reviewed in this section. Besides, in order to startup the whole project, a review towards the workstation has been made specifically for laptops because the development duration could be vary based on the specification of the workstation which is the laptop or the computer itself. In this case, the model of the laptop used in this project is A556U with Windows 10 preinstalled in the system hosted by an Intel i7 processor which is considered as a powerful workstation to be used to complete the project. Besides, the details of the Raspberry Pi SBC has also been discussed to select the optimum low cost SBC to complete the project. Based on the discussion made, the main reason that Raspberry Pi SBC has been chosen as the main SBC to run the whole system to provide the visual monitoring services is because of the compact design of the board and the power supply requirement is much lower compared to modern computers. Not only that, with the support available from the same organization, the Raspberry Pi OS makes a great component to be used with the Raspberry Pi SBC since the Raspberry Pi OS is meant to be the main software that manage the resource avaiable from the Raspberry Pi SBC. Furthermore, since this project involves with the deployment of new subsystem to improve the proposed system, the research towards Arduino microcontoller has been made and discovered that microcontroller is able to fit the requirements in this project by acting as the controller to control the electronic components according to the script compiled into the microcontroller. Moreover, to complete the proposed system, the required programming tools have also been studied in order to improve the system development quality. In this case, the programming languages that would be used in this project are Python programming language and Arduino code. Last but not least, the topic regarding CNN has also been studied to understand how deep learning works in recognition and detection. In general, a CNN model works by first receiving the input images and then perform the learning process in feature extraction sector, then the classification sector would be responsible to act as the function to perform the object classification.

2.2 Review of the Existing Systems

In the previous section, technologies review has been made specifically for the required technologies such as the Raspberry Pi, Arduino and the programming languages. Before this project was being carried out, there are similar systems being developed and implemented in order to solve the existing issues that have been affecting the quality of aquaculture farm in

terms of monitoring processes. For that reason, the following section would be discussing and reviewing the existing systems that have been developed by many researchers and developers.

2.2.1 Paper 1

First but not least, the first paper that would be reviewed is entitled "Low-shot Visual Recognition by Shrinking and Hallucinating Features" written by Hariharan and Girshick. Besides, according to the paper [17], both of the authors have mentioned that there are many modern recognition models such as Convolutional Neural Network (CNN) and Fast RCNN require a lot of labeled datasets during the training process for developing a better performance deep learning module in terms of accuracy performance. However, those of the datasets available used for training recognition models are usually consist of only a fixed set of classes for specific object. Moreover, some people may develop and deploy their recognition models in various occasions for object recognition purposes. Due to that, there are chances that the system would be encounting with the problems where the developers train their recognition models for recognizing specific novel classes with a small number of training sample. With that, it could affect the overall recognition accuracy of the recognition model. Therefore, both of the authors have proposed their system to overcome the problem as mentioned earlier.

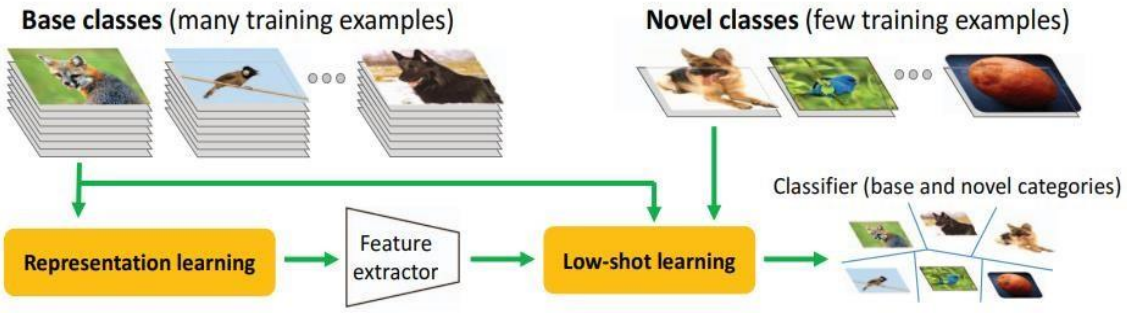


Figure 2-2-1-1 Diagram of Proposed System

Source : [17]

According to Figure 2-2-1-1, it is the visual representation of the proposed system developed by both of the authors for developing the low-shot visual learning system. Furthermore, in order to overcome with the problem as mentioned earlier, they have designed their system to allow the recognition model to generalize the object to be recognized and achieve a higher accuracy performance even if the model is being trained with very few training sample from the novel category. Based on Figure 2-2-1-1, it is shown that the system involves 2 learning processes which are the representation learning process and the low-shot learning

process. The first learning process, which is the representation learning process involves with training the recognition model with a fixed number of category set denoted as C_{base} with a dataset *D* that contains a large number of training sample from the existing categories in C_{base}. After completing the first training process, both of the authors have stated that the parameters of the feature extractor would be configured according to the *D* and begins the second training process which is the low-shot learning process. During the process, the system would provide a set of categories of the mixture of the base categories with the novel categories for the learner model. In fact, both of the authors have mentioned that the low-shot learning process is only able to access the training samples for n times where $n \in \{1, 2, 5, 10, 20\}$. In addition, the feature extractor used in the proposed system was developed based on ResNet-10 architecture and the final result shows that they were able to achieve 78.5% of accuracy performance where n = 20.

After reviewing the proposed system, there is a worth mentioning advantage that the proposed system has. As mentioned earlier, there are 2 learning processes involved in the proposed system. Based on the proposed system, a large number of dataset would be used to in the first learning process while the feature extractor's parameters would be configured based on the dataset used earlier. After that, the generated result would then be fed into the low-shot learning process. By doing so, the final recognition model could achieve a high accuracy performance without the need of execive dataset. But, based on the paper, a stronger base classifier is better than most of the weaker classifiers in generalizing objects. For instance, ResNet-10 is a weaker classifier compared to a stronger classifier such as ResNet-50. According to the proposed system, the performance of the proposed system is still not achieving the peak accuracy result. Therefore, it is highly recommended to use a better and stronger classifier such as ResNet-152.

2.2.2 Paper 2

Next, the second paper that would be reviewed is prepared by Jalal, et al entitled "Fish detection and species classification in underwater environments using deep learning with temporal information". Based on the authors, they have stated that many marine biologist and conservationists are practiced to perform fish sampling automatically without destruction to reduce the labour cost and the required time to achieve the result from manual sampling. Because of that, there are systems being studied and developed for implementing the non-destructive sampling, and automatic fish detection and species classification in underwater.

However, there are a lot of factors that have caused the developers difficult to implement such system. According to the paper, the factors as mentioned earlier include the environment lighting condition, water murkisness, high similarity of the characteristic of the aquatic entities such as shape among different species. Besides, the paper has also shown a few examples of the underwater environment with poor visibility as shown in Figure 2-2-2-1. Due to that, those factors could affect many detection system in terms of accuracy performance and this has also become the motivation for the team to overcome the factors as mentioned by proposing a system.

Figure 2-2-2-1 Example of Images Extracted from UWA Dataset Videos

Source : [18]

Based on the authors, to overcome the factors as discussed earlier, the team has proposed system that perform detection and classification by utilizing a motion detection system with a deep CNN module, specifically the YOLO architecture for detecting the movement of fish entities in different dynamic backgrounds. Below figure shows the framework of the entire system.

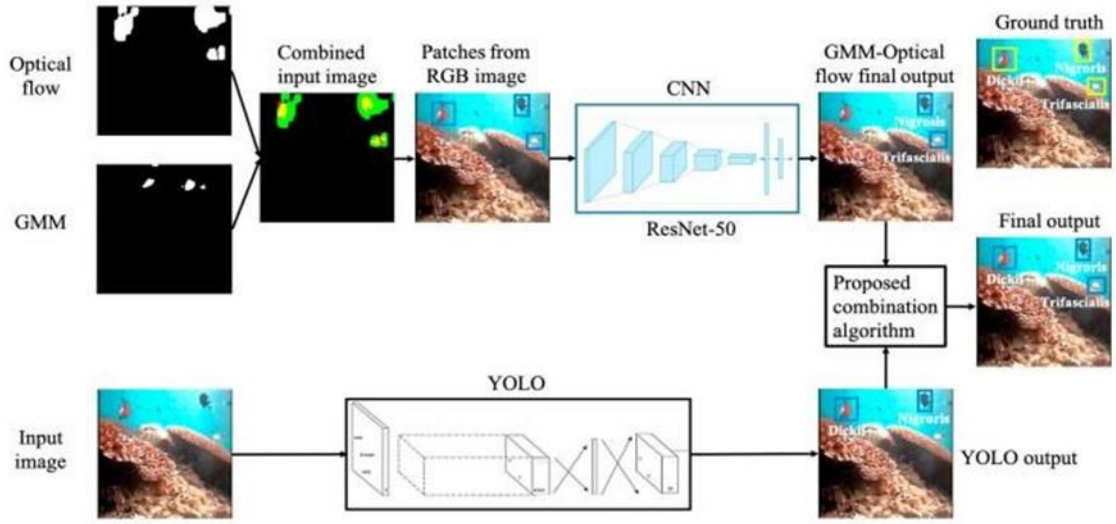


Figure 2-2-2-2 Diagram of Proposed System

Source : [18]

Moreover, according to Figure 2-2-2-2, the entire system consists of 2 different subsystems and they are the motion detection system and the deep CNN detection system. Based on the paper, it is mentioned that the motion detection system is designed and developed by merging two different models such as the optical flow and the Gaussian mixture models (GMM). Meanwhile, the deep CNN is designed and developed based on the architecture called "You Only Look Once" (YOLO) to perform the detection and classification of the species.

In the paper, it is mentioned that the proposed system would start to perform the necessary services by first running hybrid of the optical flow and then GMM model to start the motion detection system and generate the output based on the received input frame. With the result generated by both system, the result would be combined and be fed for blob analysis. After analysis process, the result would then be generated and fed into the CNN classifier develop based on ResNet-50 architecture to classify the fish entities in an image frame and generate the detection output. With that, the first half of the system has completed the overall process. Next, the YOLO model would take its turn to begin the detection and classification based on the input image received and create the recognition result. With the end results generated by two different subsystem, here is where the main part of the proposed system generate the end result by combining the outputs from the first and second subsystem with the algorithm developed by the team. Based on the paper, the team has trained their proposed system using 2 different fish datasets named LCF-15 and UWA for comparison purposes. After the training process, the proposed system is being evaluated and discovered that the proposed system was able to

achieve detection accuracy of 95.47% when using the LCF-15 dataset and the proposed system also has achieved 91.2% when being trained with the UWA dataset. The figure below shows the example of the result generated by each different subsystem along with the combined output.

Figure 2-2-2-3 Example Result from Subsystems and the Combined Result

Source : [18]

After reviewing the propsed system, there is a worth mentioning advantage that should be discussed. Based on Figure 2-2-2-2, the system is designed by combining two different subsystems. One of the subsystem is developed for motion detection powered by the GMM and the optical flow model, and then passing the generated result to the ResNet-50 deep CNN model. The second sybsystem is the YOLO deep CNN model. With such design, different outputs generated by different subsystems could be combined and create a more accurate result just like the example as shown in Figure 2-2-2-3. In short, the proposed system is able to create a more accurate recognition result by combining the results obtained from two different deep learning models. Yet, there is still a problem that should be concern in this proposed system. According to the authors, the proposed system requires a device with strong computational power due to the design of the system where it consists of complex machine learning tools. Therefore, this approach is not suitable to be deployed in devices with low computational power.

2.2.3 Paper 3

In the previous paper, the advantage of the proposed system proposed by Jalal, et al is that the design of the system is capable of generating an excellent result in detection and classification. However, due to the design of the system framework, the proposed system is not suitable to be used in a low computational computer. In this section, the paper prepared by

Chang et al entitled “Deep Learning for Object Identification in ROS-based Mobile Robots” shall be reviewed. Besides, according to the authors [19], many traditional service robots are designed with poor cognitive services such as voice and image identification. Because of that, this has become the motivation for the team to perform the necessary researches and the development of a ROS-based mobile robots powered with deep learning module to serve object identification services. In addition, Figure 2-2-3-1 describes the overall design of the architecture of the proposed system.

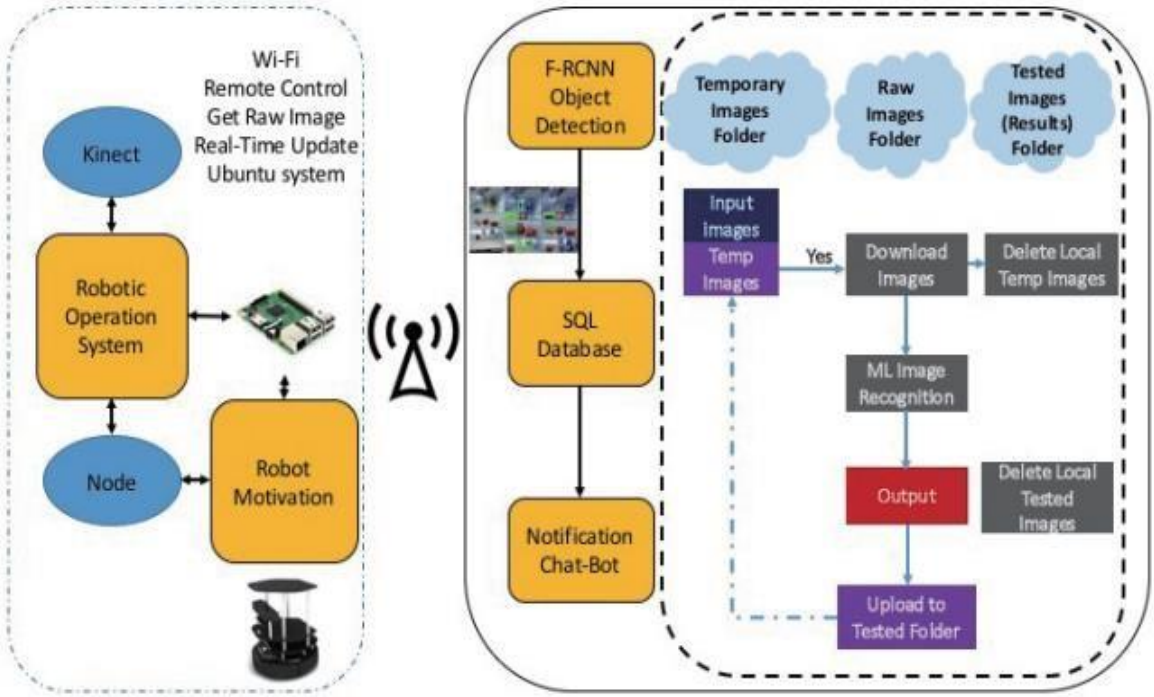


Figure 2-2-3-1 Architecture of The Proposed System

Source : [19]

According to the authors, the object recognition and detection module is the main key problem in the area of cognitive services to implement. For that reason, the team have decided to solve the problem by building a ROS-based robot that works with a Faster R-CNN algorithm based deep learning model. Besides, the authors have also mentioned that Faster R-CNN performs greatly in terms of the efficiency in generating region proposal with high accuracy. In addition, a Faster R-CNN system is developed that involves with Fast R-CNN and region proposal network (RPN). Speaking of Fast R-CNN, it is an improved model compared to traditional R-CNN.

Firstly, based on the team, they have began their project development in setting up a Raspberry Pi for their proposed system to work as the control kernel for the mobile robot that

runs with Ubuntu Mate operating system (OS). Besides, the authors have mentioned that the proposed system involves with capturing raw images of surrounding environment. For that, the team has installed a Kinect sensor into the proposed system. Once the system captured images, those images would be used as the input for the Faster R-CNN model for object detection purposes. While performing the detection process, the system would also store the output of the detection into a SQL database. To store the data into the database, the team has implemented the feature by configuring the communication between the robot and the database using Wi-Fi connection. Furthermore, according to the authors, they have stated that they have built a Faster R-CNN model based on the VGG-16 architecture. In the same time, they have stated that computing time consumption is one of the most important element to be concerned when developing a real-time application. Because of that, the team has utilised the Compute Unified Device Architecture (CUDA) to increase the system processing speed. Besides, CUDA is an application programming interface (API) specifically designed for developers to use CUDA-enable GPU for intensive task processing. In short, the API is the key in this project to speed up the processing time of the proposed system.

Based on the design of the proposed system in this paper, there are a few ares that should be mentioned and discussed. First and foremost, based on Figure 2-2-3-1, the overall design of the proposed system comes with two individual subsystems with different purposes. The authors have mentioned that the first component of the system that would start the overall process is the robot itself which is controlled by the Raspberry Pi. The subsystem would first capturing images, and then send the input images for the detection model to generate the detection output. After generating the detection output, the output would be stored in the SQL database via Wi-Fi connection. Furthermore, since the hardware that is responsible to handle the detection model processing utilises the CUDA API, this allows the Faster R-CNN architecture to perform greatly in real time object detection. But, there are a few disadvantages in using this system. Firstly, based on the design of the system, the images taken by the robot would be stored into the database via Wi-Fi. In other words, if the Wi-Fi connection is down, the detection system would not work without the input images from the robot. Besides, the authors have also mentioned that they were required to use the CUDA API to increase the efficiency performance of the detection system. In other words, CUDA API is only available in certain GPU produced by NVIDIA and the cost of the GPU is not cheap. In shorts, the disadvantage of the proposed system is that the system still require a high cost and high computational power computer to run the detection system.

2.2.4 Paper 4

In this section, it will be the final paper which is the fourth paper to be reviewed is prepared by Curtin and Matthews which is entitled "Deep Learning for Inexpensive Image Classification of Wildlife on the Raspberry Pi" [20]. According to the authors, they have stated that many commercial cameras specifically made for wildlife observation are costly and the cost of those cameras ranged between $100.00 to $400.00. Besides, they have also mentioned that those camera may not necessarily suitable to be used in remote environment without cellular signal. Not only that, those cameras are usually being used to take pictures and works whenever there is something triggering the camera. Because of that, the owner of the camera needs to shift through all the pictures taken manually to check if the pictures taken satisfy the owner. Because of that, the authors have taken this opportunity to research and develop a low cost image classification system by utilising Raspberry Pi, specifically Raspberry Pi 3B+ SBC. In order to allow the Raspberry Pi to capture images, they have installed a Raspberry Pi based camera module in the Raspberry Pi SBC. With this, the overall hardwares only cost around 50$ which is cheaper than those expensive cameras. Based on the comment from Curtin and Matthews [20], they have chosen to use this particular model of the SBC in this project is because most of the older versions of Raspberry Pi lack of the computational power to handle the image processing process which is quite heavy. Also, since image processing is a heavy process, most of the developers would train their CNN model using a computer with higher computational power rather than training the model on the Raspberry Pi. After training the model, the model would then be embedded into the Raspberry Pi and starts the image processing services. During the development process, both of the authors developed their recognition model using available libraries such as TensorFlow and Keras because those libraries do not cost money and supported in the Raspberry Pi. Furthermore, the model developed by the authors is a basic CNN architecture which consists of a few 2-D convolutional layers. In the first layer, there are 32 feature layers with a 3 x 3 feature detector with 1 stride length. As for the second layer, there are 64 features layers with a 3 x 3 feature detector with a ReLU (rectified linear unit) activation function. Also, the authors have added a max pooling into both of the layers as mentioned earlier to reduce the total feature set. In the third layer, they have designed it to be flattened and to act as the input for a dense layer (or linear layer) with 128 input units along with a ReLU. Not only that, the architecture of the CNN model also has a dropout function to drop 20% of the inputs and the remaining inputs would be passed into the last dense output layer. Last but not least, the authors have added softmax activation function into the last layer of the CNN

model to support the different dimensions that represent the object intended to be detected and classified. In this project, there are 3 dimensions or classes to be detected and classifed which are the “snow leopard”, “human”, and “other”. To train the model, the authors have generated a total of 3,598 images to use it as the input to train the model. Among the images, there are 1,262 images of humans, 1,568 images of snow leopards and 768 images of natural background for the dimension called “other”. Once the training process is completed, they have evaluated their detection model with some pre-downloaded images. Below tables show the result of the accuracy of classification for each categories.

	Snow Leopard	**Human**	**Other**
Accuracy (%)	97	99	96
Precision (%)	99	100	85
Recall (%)	94	96	99

Table 2-2-4-1 Accuracy Test With Pre-downloaded images

Source : [20]

	Snow Leopard	**Human**	**Other**
Accuracy (%)	74	77	72
Precision (%)	57	95	63
Recall (%)	97	41	53

Table 2-2-4-2 Accuracy Test With Live images

Source : [20]

After reviewing the paper, it is discovered that there are a few aspects should be discussed here in order to understand the advantages that the proposed system could provide. Besides, this is necessary because such deployment could be used in many sitaution including this project. First of all, according to Curtin and Matthews [20], both of them aimed to complete their project by building a low cost object classification system. Also, according to the cost estimation [20], the overall cost needed to prepare the required hardwares is 49.50$. In short, this is very suitable for many people especially university students that would like to study or develop an image classification system without spending too much on hardware preparation. Table 2-2-4-3 shows the overall cost of the hardwares required by the authors.

Item	Cost
Raspberry Pi 3B+	$35.00
Raspberry Pi Camera Module	$5.00
16 GB microSD Card	$4.50
4 AA Batteries	$2.00
AA Battery Holder with Micro USB Cable	$3.00
Total	**$49.50**

Table 2-2-4-3 Estimated Hardware Cost

Source : [20]

Besides, according to the authors [20], it is mentioned that the deep learning model used in the proposed system is only about 19MB in size. In other words, this is also an advantage of using the same approach to build the deep learning model if storage space is very limited to users. However, even though the proposed system has its advantages, it also has its disadvantages too. Firstly, based on the proposed system architecture, both authors have mentioned that once the system is triggered and captured an image, it will send the image to the user automatically. Also, since the power supply of the Raspberry Pi SBC comes from the 4 AA batteries, the image capturing process might deplete the battery life in short amount of time if there is a lot of transmission occured. Not only that, based on the authors, they are using the Raspberry Pi based camera module for image capturing purposes and the quality of the pictures taken by the camera itself is lower compared to the image dataset used to train the model during the development process. In simpler terms, the classification model may not be performing at its best in terms of accuracy result due to the lower image quality.

CHAPTER 3

System Model

3.1 System Design Diagram

In this chapter, the overall design of the proposed system in this project would be shown. Besides, there are 3 diagrams in the following section and they are the system architecture diagram, use case diagram and description and activity diagram.

3.1.1 System Architecture Diagram

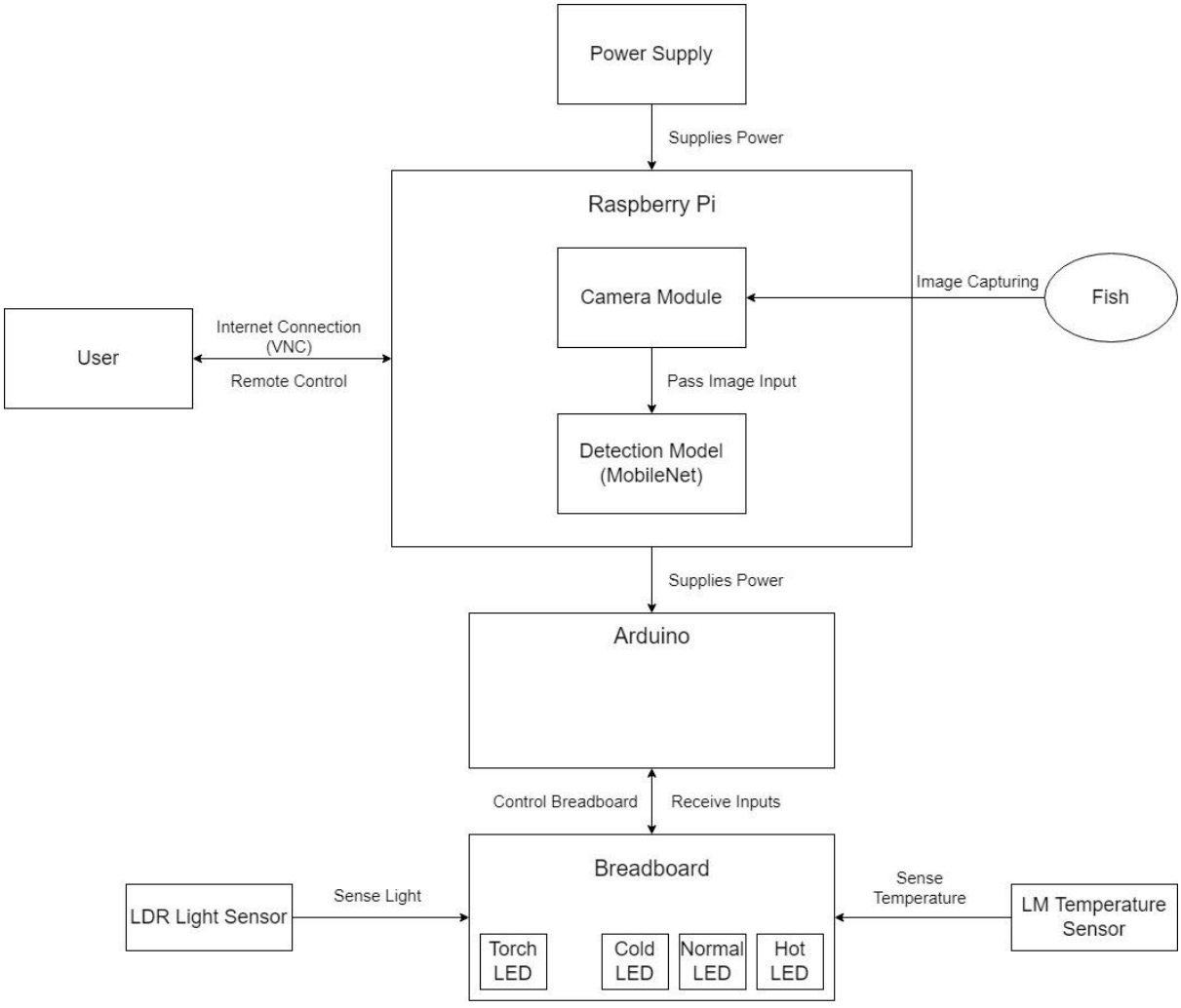


Figure 3-1-1-1 System Architecture Diagram

Based on the diagram above, it is illustrating the design of the proposed system used in this project. Again, based on the diagram, the proposed system is mainly made up of two

components. The first is the Raspberry Pi, and the second is the Arduino. As discussed in the previous chapter, the main difference between the Raspberry Pi and the Arduino is that Raspberry Pi is a microcomputer while Arduino is a microcomputer. In this project, the Raspberry Pi is responsible to perform the fish detection system. To perform the process, the Raspberry Pi would first capture the fish images using the camera module installed into it. Then, the images would then be passed to the detection model to generate the detection output. To use the system, users are allowed to use it directly at the machine site or via remote access. Next, the Arduino would be responsible for providing the information of environment temperature and illuminance. Technically, Arduino is capable of controlling many types of devices such as LEDs and sensors with the help from a breadboard to connect those components together. Based on the design of the system, there are 2 sensors which is the LM temperature sensor and the LDR light sensor. Besides, each sensors would begin to perform the sensoring task and provide the information to the Arduino controller. As for the LDR sensor component, it is one of the important elements in this system to indicate the illuminance of the environment. Moreover, if the environment reaches specific illuminance level, the "Torch LED" would be turned on indicates the environment is dark and additional light is required to allow enough lights for the Raspberry Pi camera to continue the detection services. Otherwise, the "Torch LED" would be turned off if the environment's light level is in a good condition. Furthermore, some fishes prefer to live in a colder or warmer environment. Because of that, the temperature monitoring subsystem is implemented. Based on the design of the diagram, the way how the subsystem works is fairly simple. For example, if the temperature sensor sense a very cold temperature, the Arduino would trigger the "Cold LED". In short, this specific model triggers specific LED based on the temperature sensed by the temperature sensor. Also, as discussed in the previous chapter, Arduino does not consume a lot of power in order to operate normally. For that reason, the power supply could be easily solved by connecting the Arduino to the Raspberry Pi directly to act as the power supply. In short, that is how the proposed system works in general and more details would be explained in the subsequent chapters.

3.1.2 Use Case Diagram and Description

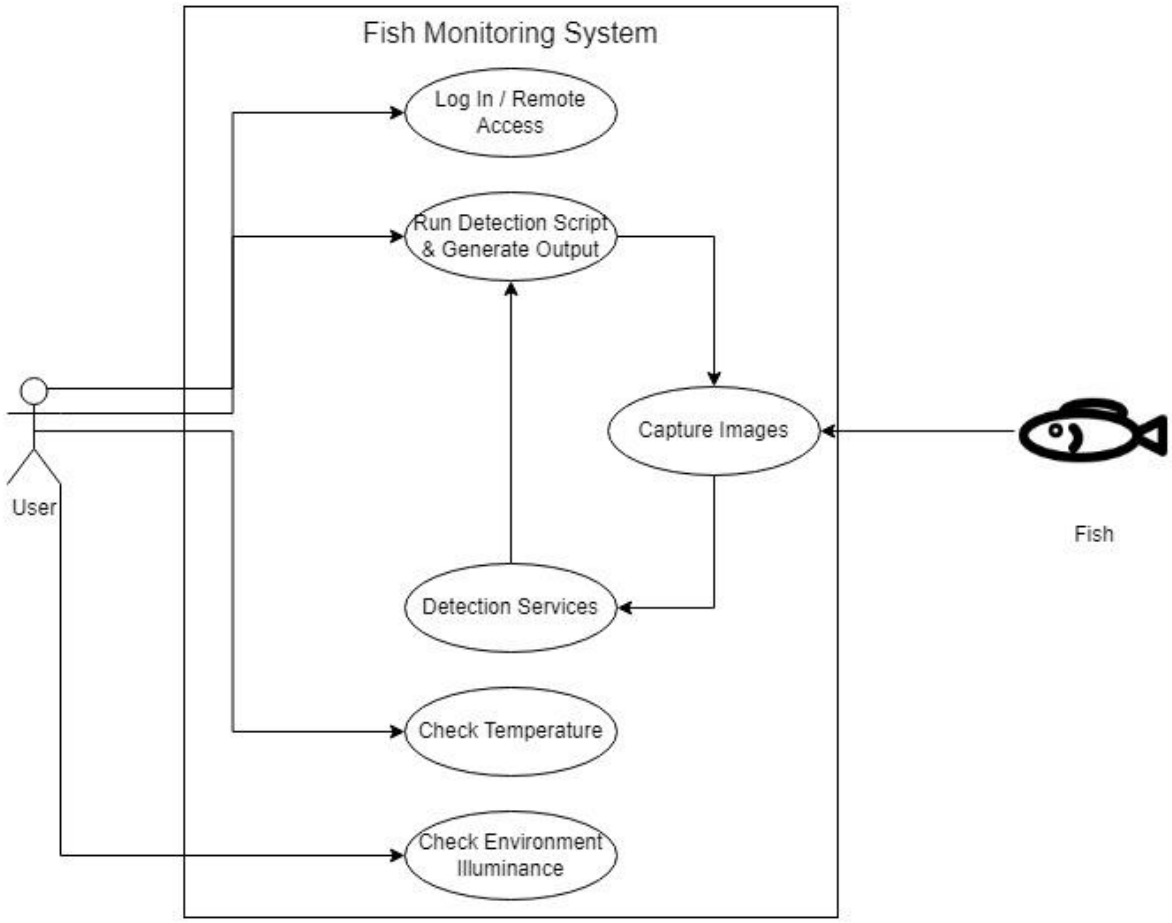


Figure 3-1-2-1 Use Case Diagram

Based on the use case diagram, there are total of 6 activities that the proposed system could process and 4 among the activities could be performed and controlled by users. First of all, this system is designed to be usable directly at the machine site or in a distance. Therefore, users are allowed to gain remote access to the system via network connection through an software application known as VNC if users wish to use the system from a distance. Besides, once the users have gained remote access to the system, they could proceed to use the system to run the detection script or deep learning module to detect fish entity from the images fed and captured by the camera. Also, users are also allowed to monitor the environment temperature and illuminance when they are using the system at the machine site and the specific details would be discussed in the following chapter.

3.1.3 Activity Diagram

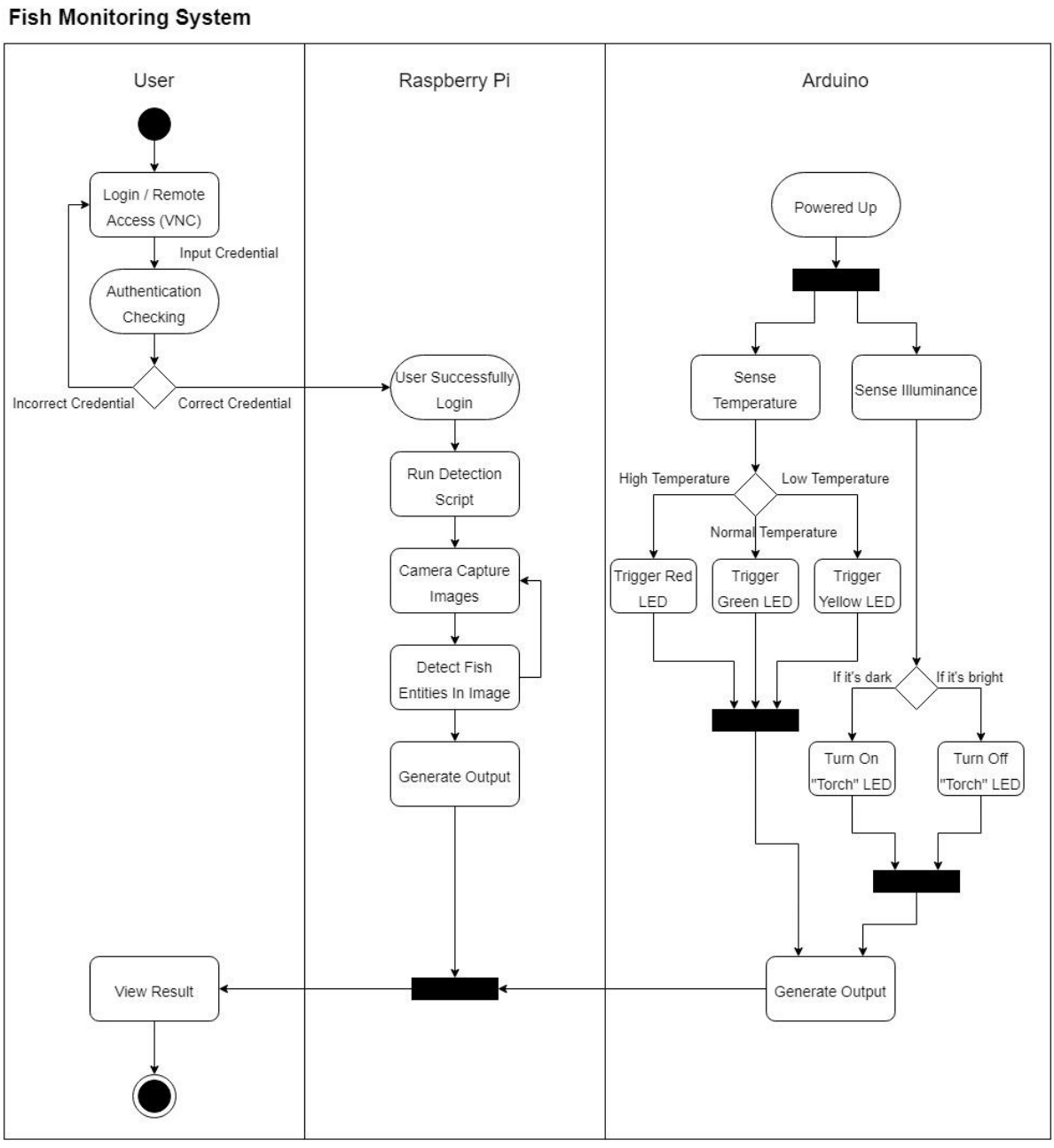


Figure 3-1-3-1 Activity Diagram

Based on the diagram above, it is showing the activity diagram of the proposed system. As mentioned earlier, users are allowed to use the system either by using it directly at the machine site or by remote access. In this diagram, it shows how a user could use the system via remote access. First of all, user may use any computer such as a laptop or desktop and gain remote access to the Raspberry Pi via VNC. To do that, users are required to enter the credential to login into the Raspberry Pi. After providing the correct credential, users are able to proceed to

use the Raspberry Pi to perform the detection system and get the detection result. As for the Arduino section, the component would perform the temperature and illuminance monitoring process without the needs of input from user and the generated outputs would be shown with LED light indicator as mentioned earlier in the 3.1.1 section. After that, users would be able to see the result and terminate the whole process or continue using the system according to their usage requirement.

CHAPTER 4

System Design

In this chapter, the design of the system including the details of the system implementation would be discussed as well. This chapter would be the section that describes the main content of the whole research paper. Therefore, the following would be discussing the necessary details of the proposed system's implementation.

4.1 System Block Diagram

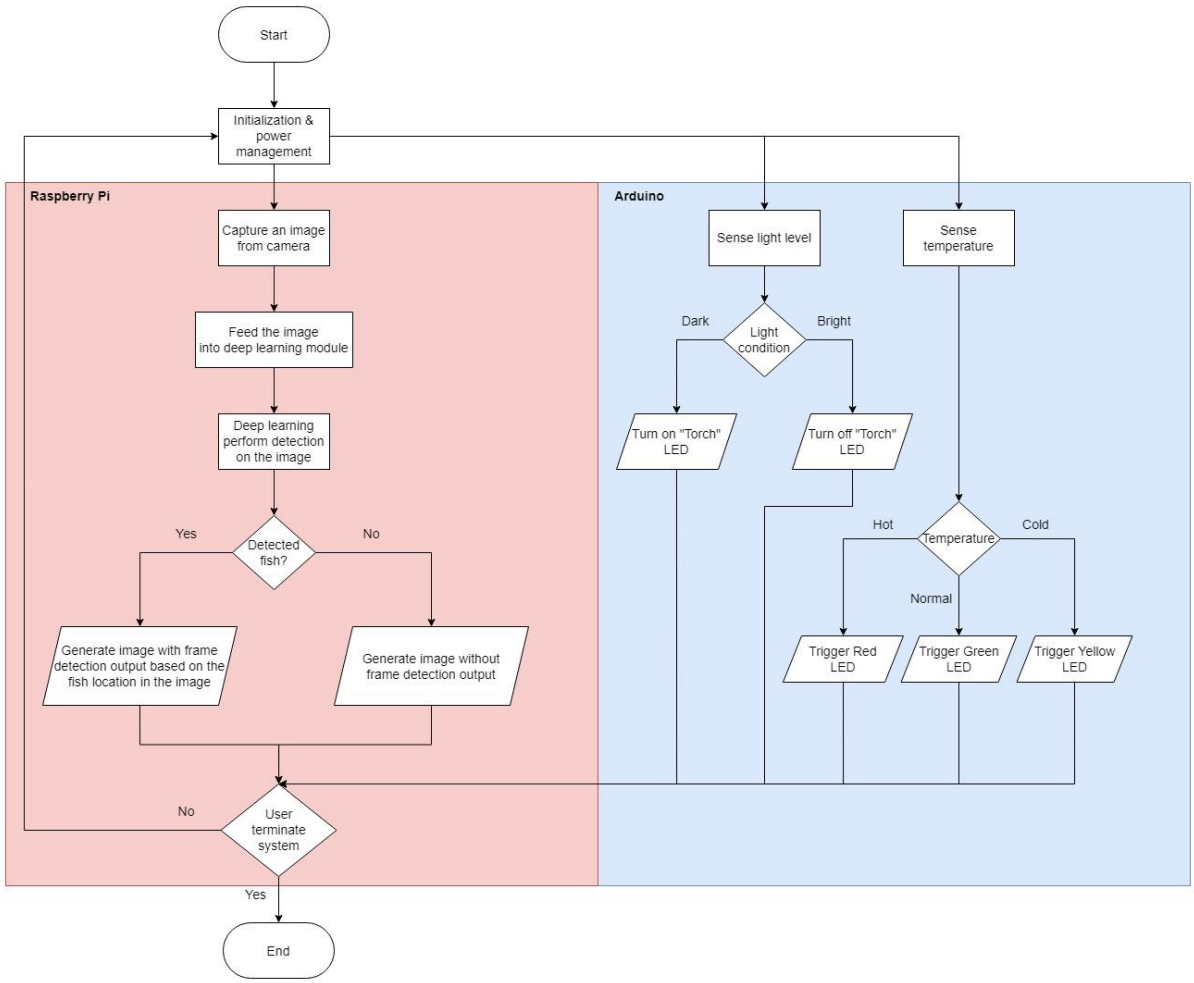


Figure 4-1-1 System Block Diagram

Based on Figure 4-1-1, it is showing the system block diagram that describe the overall processes involved in the proposed system. In the proposed system, it is mentioned that there are two subsystems combined to form a new system. In this case, one of the subsystem is the

fish detection system with deep learning module powered by the Raspberry Pi which is indicated with red area in the diagram and another subsystem is powered by the Arduino microcontroller which is indicated with blue area in the diagram to monitor the environment temperature and light level to trigger the specific LED for indicating the environment situation.

When the user has started the system, the whole system would first perform the necessary initialization for the deep learning module and power management for the Arduino board to start the program configured into it. In the Raspberry Pi, it would begin the fish detection system by capturing images using the camera module installed in the system once the initialization process is completed. After that, the captured image would be fed and passed to the deep learning module to perform the detection services. Besides, if the deep learning module detects a fish entity in the image frame, the deep learning module would generate the image output with labeling according to the location of the fish entity in the image frame. Otherwise, the deep learning module would just output the image frame without any labels. Meanwhile, as for the Arduino, the board would start its tasks whenever it has received the power supply from the Raspberry Pi. Based on the diagram above, the Arduino would begin its task by monitoring the environment light level and temperature by using the sensors as mentioned in 3.1.1. As for light level monitoring process, if the light sensor senses that the light level of the environment is dark, it would then trigger the "Torch" LED to indicate additional light is required. Otherwise, the "Torch" LED would not be trigger to indicate that the environment's light level is considered bright. For the temperature monitoring process, whenever the temperature in the environment is high, the Red LED would be triggered. Besides, if the surrounding temperature is very low, the Yellow LED would be triggered. Otherwise, the Green LED would be triggered to indicate that the temperature of the environment is in good condition. Furthermore, these processes would be repeated continuously until the user terminate the whole system.

4.2 System Components Specifications

In this section, all of the components and equipments would be listed and explain the reason for choosing the particular component. The table below shows the equipments used in this project to implement the proposed system.

No.	Item Name
1	Laptop

2	Raspberry Pi 3 Model B+
3	Raspberry Pi Camera Module
4	5V/3A Raspberry Pi 3 Power Adapter
5	16GB Memory SD Card
6	Ethernet Cable
7	Arduino Uno R3
8	Breadboard
9	USB A-to-B Cable
10	LED (2 Greens, 1 Red, 1 Yellow)
11	Jumper Wires (male-to-male x 11)
12	LM35 Temperature Sensor
13	LDR Light Sensor
14	Resistor 220 Ohm x 5
15	Protective Case (Optional)

Table 4-2-1 List of Equipments

As mentioned in the previous chapter, it is mentioned that a Raspberry Pi would be used as the main component to perform the fish detection system. Besides, the specific model of the Raspberry Pi that would be used in this project is the Raspberry Pi 3 Model B+. The details of the board's specification and the reason of using the board is explained in chapter 2 under section 2.1.2. In addition, since the proposed system would need to capture the images of fishes, a Raspberry Pi camera module is used and installed in the Raspberry Pi board. To power up the Raspberry Pi board, the board requires at least 5.0V/3A of power supply. Because of that a 5V/3A Raspberry Pi 3 Power Adapter is used. Also, since the SBC board does not have storage built in, it would require a memory sd card to store and boot the OS. So, a basic 16GB Memory SD Card is enough for this project since the SD card would not be used for storing a lot of data. Although the items as mentioned is enough to run the deep learning module, it is still recommended to have an extra protective case for the SBC board. Moreover, in order to implement the deep learning module and every basic setup for the Raspberry Pi, a laptop or a computer is still one of the important and required element in this project and since the system setup involves internet connection, an ethernet cable is required. Next, the Arduino that would be used in this project is the Arduino Uno, more information about the specification is explained in 2.1.4 and the reason this model of microcontroller is used is because it already has

all of the basic functionality that could fulfill the requirements in this project. Besides, the components that would be used in designing the circuit is breadboard, USB A-to-B Cable, some LEDs, jumper wires, LM35 temperature sensor, LDR light sensor, and some 220 Ohm resistors.

4.3 Circuits and Components Design

After completing the necessary studies and equipment preparation, the initiation to develop the fish monitoring system for this project has been taken. Besides, the system implementation begins with some important plannings and preparation that could not be ignored. Based on the Figure 3-1-1-1, it is illustrating the overall design of the architecture of the fish monitoring system. To understand further, the following will explain the steps taken to build the fish monitoring system.

4.3.1 Software Requirement & Preparation

After preparing all of the required hardwares, there are some of the softwares are required to download and install in order to implement the system as well. Therefore, the following are the softwares required.

1. Python Programming Language
2. Anaconda with Jupyter Notebook
3. Tensorflow Library
4. LabelMe
5. Raspberry Pi OS Imager

In this project, the main task to be completed is the deep learning model for detecting fish entities from an image. To develop the deep learning model, the main programming tool that would be used is Python. After installing Python programming language, the next software that is required in this project would be the Anaconda. Besides, Anaconda is a great platform that works with Python and it is popular in data science or deep learning development. Also, to start writing Python code, the Jupyter Notebook should be installed in the Anaconda platform. Next, to start the deep learning development, there is one library required to be installed which is known as the Tensorflow library. In addition, Tensorflow is an open source platform used in building machine learnings and this platform is developd by Google. Moreover, before developing the deep learning module, image dataset is needed for training the deep learning module once it is developed. To prepare the dataset, a tool called LabelMe would be used in

this project to prepare the dataset for training the deep learning module. Last but not least, after training the module with the dataset, the deep learning module is ready to be embedded into the Raspberry Pi. However, the Raspberry Pi still require a tool to setup and configure the basic configuration on the Raspberry Pi board in order to allow it to boot properly. In this case, the Raspberry Pi OS Imager is the tool that could help in system configuration.

4.3.2 Dataset

As mentioned earlier, to develop the deep learning module, it requires image dataset to train the deep learning module in order to learn the features of the fishes to be detected. In final year project (FYP) 1, there were 115 of images used to train the deep learning module and there were 17 images being used for model evaluation. In the dataset, it contains various types of fishes such as a Betta fish and a goldfish as shown below.

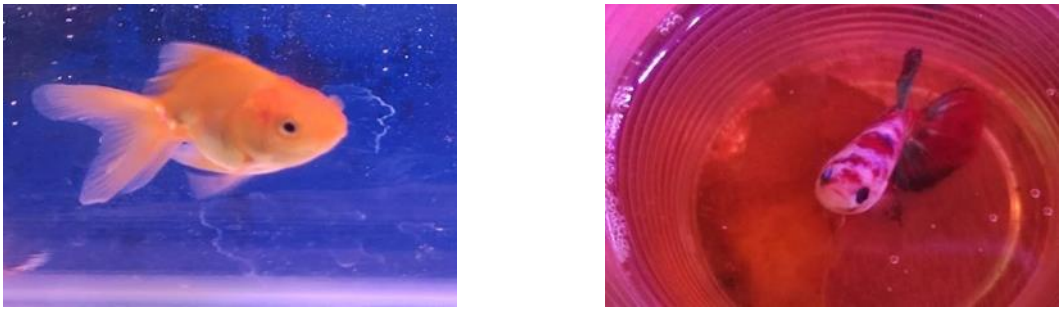


Figure 4-3-2-1 Sample Images Used in FYP 1

At the end of the project in FYP 1, the accuracy result obtained from the training was about 18% at best. Based on the result, it is considered very low to properly allow a system to monitor the fish in aquaculture farm. Because of that, investigation has been carried out and a hypothesis has been made to overcome with the issue that happened in the previous project. According to the investigation made, the fish images used in the previous project are very different from each other in terms of their characteristic. For example, the skin color of a goldfish is normally gold and the skin color of a Betta fish could be red or any combination of different colors. Not only that, based on the real life practices, most developers would train their deep learning module with at least a few thousands of images obtained from some well known databases such as ImageNet database. During the development process, the access to public databases is very limited and preparing a few thousands of image dataset is very time consuming. For that reason, there is a few changes made in dataset preparation for this project. First of all, instead of using images of various types of fishes, the images of a single type of

fish is used. In other words, instead of detecting any “fish”, the fish entity that would be aimed as the fish to be detected by the proposed system is named “Golden Ryukin Calico” (calico). Logically, this action would allow the deep learning model to learn more about the specific characteristic of the fish without using too much dataset to create a higher mAP result, and this would be discussed further in the following sections. Besides, there are 124 images used for training the deep learning module and there are 8 images used for model evaluation. In total, there are 132 images used in this project which is slightly higher compared to the number of images used in FYP 1 (115 in total). Furthermore, due to limited access to public databases, the images obtained in this project was done by capturing the images of the fish from a local aquatic pet shop with permission given. The following is the sample image dataset used to train the deep learning module.

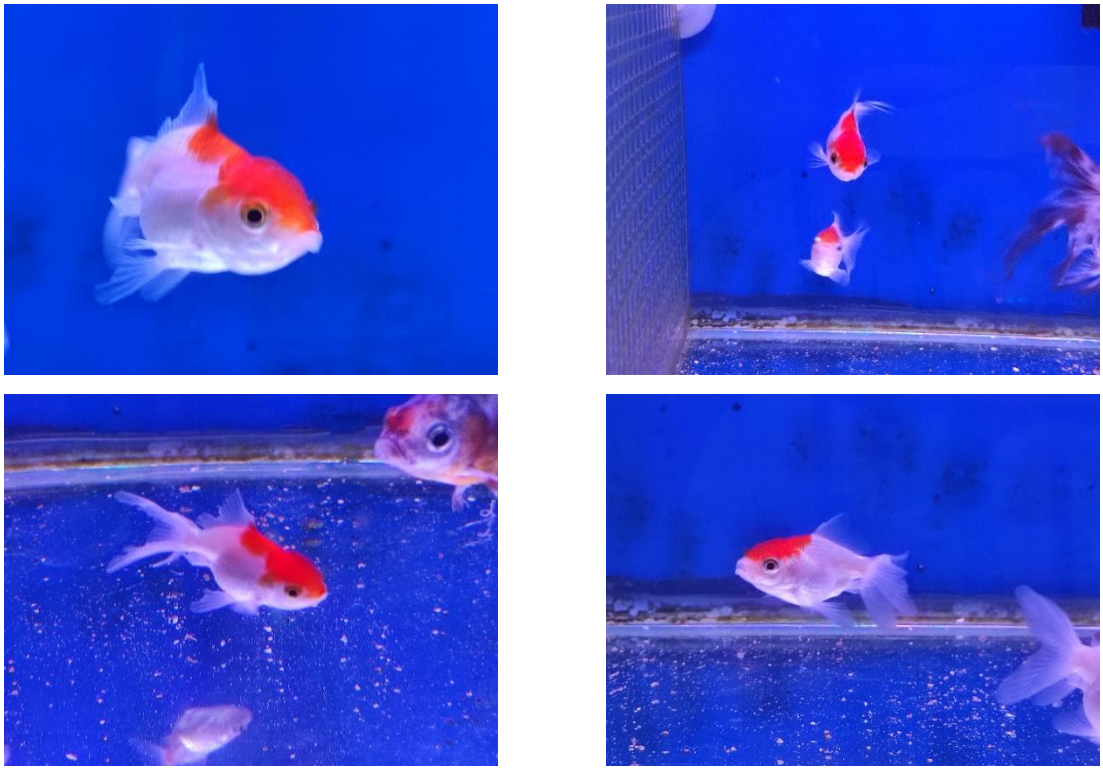


Figure 4-3-2-2 Sample Image Dataset

4.3.3 Deep Learning / Detection Model

In this project, the main component that would be developed is the deep learning module that is responsible for detecting the existence of the fish entity from images. Besides, the tools that have been used in developing the deep learning module are the Python programming language, Anaconda with Jupyter Notebook IDE, and the Tensorflow library which is explained in section 4.3.1. Based on the studies made in FYP 1, MobileNet is a very lightweight

CNN model that is best to be used and deployed in computers with lower computational power such as the Raspberry Pi SBC. In addition, the model of the MobileNet used in the previous project is also known as “SSD MobileNet V2 FPNLite 320x320”. In fact, it is found that there are more variety of MobileNet CNN models which is available on a GitHub page entitled “TensorFlow 2 Detection Model Zoo” and the diagram below shows a sample of the GitHub page with several different downloadable models.

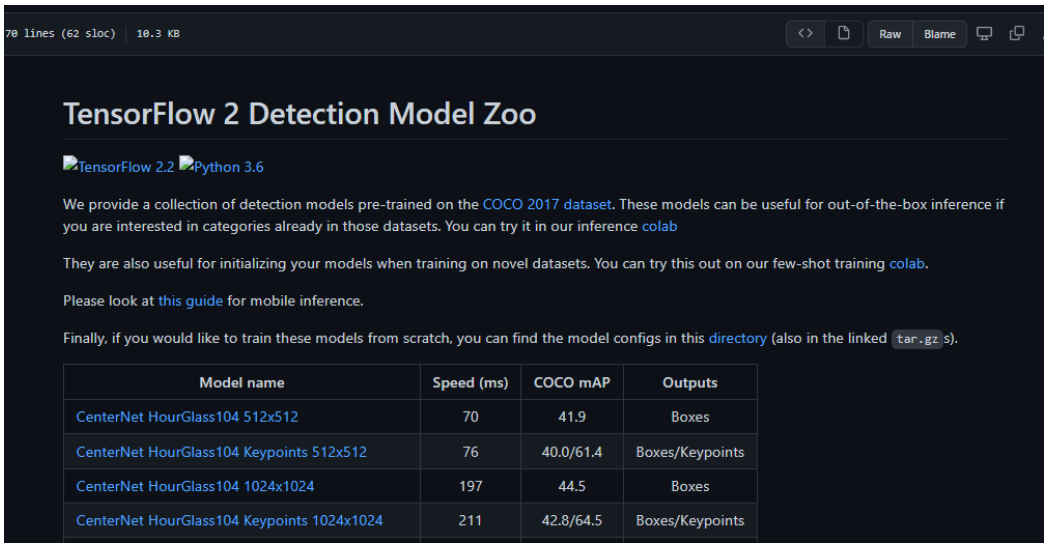


Figure 4-3-3-1 Sample GitHub Page

Source :

(https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md)

Again, based on the studies made in FYP 1, one of the best way to make a selection in choosing specific deep learning architecture for object detection is by comparing different type of deep learning architectures such as comparing the ResNet-50 architecture with the MobileNet architecture. In this project, the similar approach would still be used and there are 3 different variety of MobileNet architectures are going to be compared in terms of accuracy and efficiency performance. The table below shows the variety of the MobileNet architectures that are going to be used in performance evaluation.

No.	Architecture	Speed (ms)	COCO mAP
1	SSD MobileNet V2 FPNLite 320x320	22	22.2

2	SSD MobileNet V2 FPNLite 640x640	39	28.2
3	SSD MobileNet v2 320x320	19	20.2

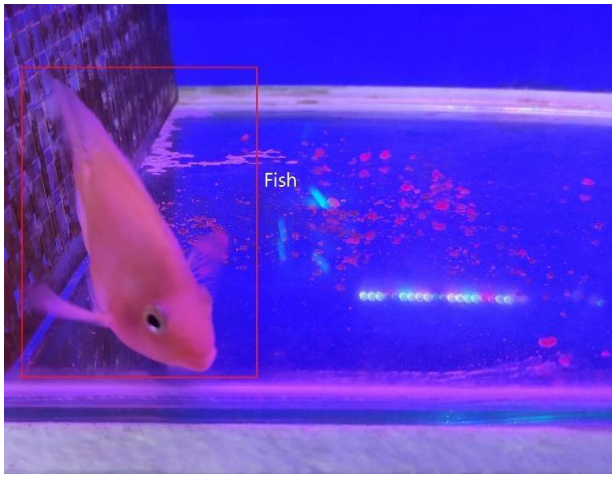
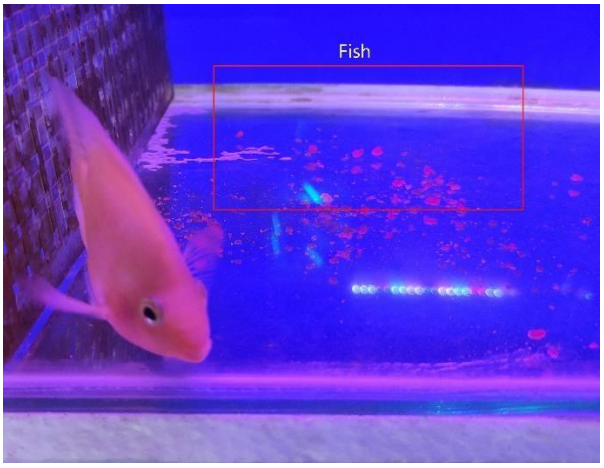
Table 4-3-3-1 List of Architecture

Source : (https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md)

Based on Table 4-3-3-1, all of the variety of the architecture performs differently from each other not only in speed but also in mAP result. Besides, the first architecture as shown in the table is the architecture that has been used in the FYP 1 and the benchmark result is 22 in detection speed and 22.2 in mAP metrics. In addition, the speed as stated in the table describes the time required by the model to refresh and proceed to perform the next detection. In other words, the lower the value of the speed, the lesser time the model requires to proceed to the next detection process. Next, the second architecture listed in the table is named "SSD MobileNet V2 FPNLite 640x640". According to the table, this particular model has better mAP value which is 39 compared to the first architecture where it only has 22. However, the downside of the architecture itself is that it is slower compared to the first architecture. Last but not least, the third architecture as listed in the table is named "SSD MobileNet v2 320x320". Besides, the mAP performance of the model is slightly lower compared to the first model which is 19. Other than that, the speed performance of this model is faster than the first architecture. In short, "SSD MobileNet V2 FPNLite 640x640" is considered as the heaviest model with highest mAP performance and "SSD MobileNet v2 320x320" is the lightest model with lowest mAP performance.

Moreover, in order to verify the performance of each different architecture, an experiment has been carried. Besides, before discussing about the findings of the experiment, there are a few terms should be explained and understand. First of all, deep learning modules are usually being trained according to a specific parameter to determine the number of iteration the model should perform to complete the training process. In this case, the term training step could be used to describe it. Other than that, a deep learning model is usually being evaluated based on these 3 metrics which are the mAP, recall and total loss. First of all, mAP refers to the average of average precision (AP) and AP is used to determine the accuracy detected for specific class. In this case, the class that is defined in this project is the calico. Also, since there is only a

single class being defined in this project, the value of the mAP and the AP should be equal. In other words, mAP is a metrics that determines how well a deep learning model could correctly identify an object. Next, recall metrics is also a very important aspect used for determining how well a deep learning model determine if the given image input should be categorised as a positive result. Besides, the reason this is important because a deep learning model may have a great mAP performance but not recall performance. Without recall performance, the overall performance would greatly reduced. For example, Table 4-3-3-2 shows the example result of the functionality of recall function. Last but not least, total loss or loss metrics could be used to represent a CNN architecture prediction error. In other words, loss metrics determines how a deep learning model works incorrectly in classification.

True Positive	
False Positive	

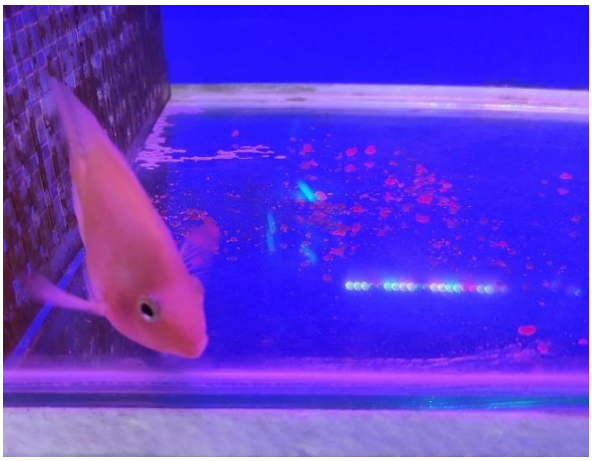
False Negative	

Table 4-3-3-2 Sample Recall Result

Furthermore, the experiment carried out in this project is similar to the experiment being carried out in FYP 1 to verify the performance of each model. In the experiment, all of the deep learning models have been deployed and trained with the image dataset as described in section 4.3.2. Before training each of the model, the batch size that has been configured is 4 and the number of training step configured is 2500 which is exactly the same setting as configured in FYP 1. Once the model evaluation is made, the result of the experiment is as shown in the following table. Based on the result in Table 4-3-3-2, the first deep learning model was able to achieve the result of the mAP metrics of around 0.82 or 82.42%. Besides, the model was also able to achieve 0.67 of recall metrics and around 0.35 of total loss. Next, the second deep learning model was also able to achieve a great result in most aspect. Based on the result obtained from the experiment, the second model was able to perform the detection services with the mAP of around 0.77 or 77.04% and recall metrics of 0.63. In addition, the total loss metrics produced by the model was around 0.32. In comparison, it is shown that the first detection model is able to perform better than the second detection model in terms of detection accuracy with the exception of total loss. Next, the third variety of the MobileNet has also been deployed in the experiment and the result obtained was far different compared to the expected result. Based on the result obtained, the mAP result is only around 0.06 or 6% and the recall metrics is only 0.09 which is quite unexpected and these metrics values are even lower than the metrics values obtained from FYP 1. Meanwhile, the total loss produced by the model is around 5.2531 which is considered very high in terms of error rate. Based on the experiment, it is concluded that the “SSD MobileNet V2 FPNLite 320x320” performs better compared to other models and the result obtained with the first model has accomplished one of the project scopes which is to develop a fish monitoring system with the performance of detection module

with at least 70% of mAP (mean average precision), and recall metrics of at least 50%. Therefore, the first model which is the "SSD MobileNet V2 FPNLite 320x320" would be used as the main detection model for the proposed system.

No.	Architecture	Training Step	mAP	Recall	Total Loss
1	SSD MobileNet V2 FPNLite 320x320	2500	0.82425	0.67	0.35511
2	SSD MobileNet V2 FPNLite 640x640	2500	0.77048	0.63	0.32764
3	SSD MobileNet v2 320x320	2500	0.06445	0.09	5.25317
4	SSD MobileNet V2 FPNLite 320x320 (FYP 1)	2500	0.1811	0.3359	

Table 4-3-3-3 Performance of Every Models

4.3.4 MobileNet SSD v2 & YOLOv5

In the previous section, the "SSD MobileNet V2 FPNLite 320x320" is able to perform well in terms of detection accuracy and has been chosen as the model to be used in this project. Furthermore, even though the experiment has shown that the MobileNet architecture performs decently, further studies suggested that there is a CNN architecture performs even better than the MobileNet architecture and the architecture is known as YOLOv5 (You Only Look Once). Moreover, based on the paper preprared by Varadharajan et al [21], the team has tried to developed three different architectures of real-time mask detection system based on YOLOv3, YOLOv5, and MobileNet SSD v2. After developing all of the models, the team has performed evaluation on each models based on the mAP metrics and frame per second (fps) which describes how fast the detection perform. According to Table 4-3-4-1, it is showing the result of performance of the three models and the result shows that the YOLOv3 models was able to ahieve mAP metrics of 54.3% which is higher compared to the YOLOv5 and MobileNet. However, the model itself is heavier than the other models hence it is not suitable to deploy such architecture in real-time application. Besides, the result also shows that the performance of YOLOv5 in detection accuracy and speed is better than the MobileNet SSD v2. Therefore,

this has become the motivation to develop a YOLOv5 model to compare the accuracy and speed performance of the model developed in this project.

Model	mAP (%)	TeslaT4 (fps)
YOLOv3	54.3	80
YOLOv5	37.6	100
MobileNet SSD v2	33.7	94

Table 4-3-4-1 Performance Analysis of The Three Models

Source : [21]

First of all, while developing the YOLOv5 model, the model itself was not avaiable on the Tensorflow zoo GitHub page. To develop the new model, a guide available on GitHub repository provided by Jocher Glenn is followed and the figure below is the sample of the GitHub page.

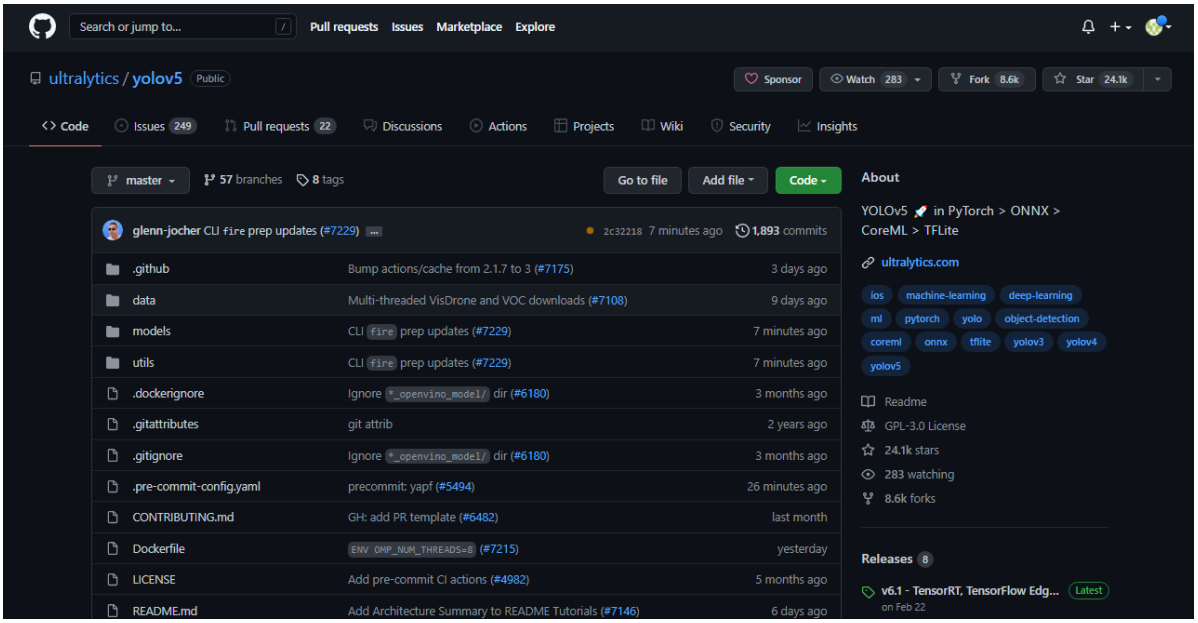


Figure 4-3-4-1 Sample of The GitHub Page

Source : (https://github.com/ultralytics/yolov5)

While training the model, the dataset used for this model is the image dataset as discussed in section 4.3.2 and the training configuration used in the process is configured to be as similar as the configuration made while training the MobileNet models. After completing the training process, the model is being evaluated and the following table shows the result of the YOLOv5 model. Based on the result in Table 4-3-4-2, it has shown that the YOLOv5 model performs better than the MobileNet in terms of the detection accuracy in mAP and recall metrics.

However, even though it is proved that YOLOv5 performs better than MobileNet, the YOLOv5 is not being deployed in this project due to limited time and personal capability to convert the deep learning model into a suitable file format to be embedded into the Raspberry Pi. Therefore, the MobileNet model would still be used in this project as the main detection model for detecting the fish entity. Figure 4-3-4-2 shows the sample detection result produced by the YOLOv5 model.

No.	Architecture	mAP	Recall
1	SSD MobileNet V2 FPNLite 320x320	0.82425	0.67
2	YOLOv5	0.86143	1

Table 4-3-4-2 Performance of MobileNetv2 & YOLOv5

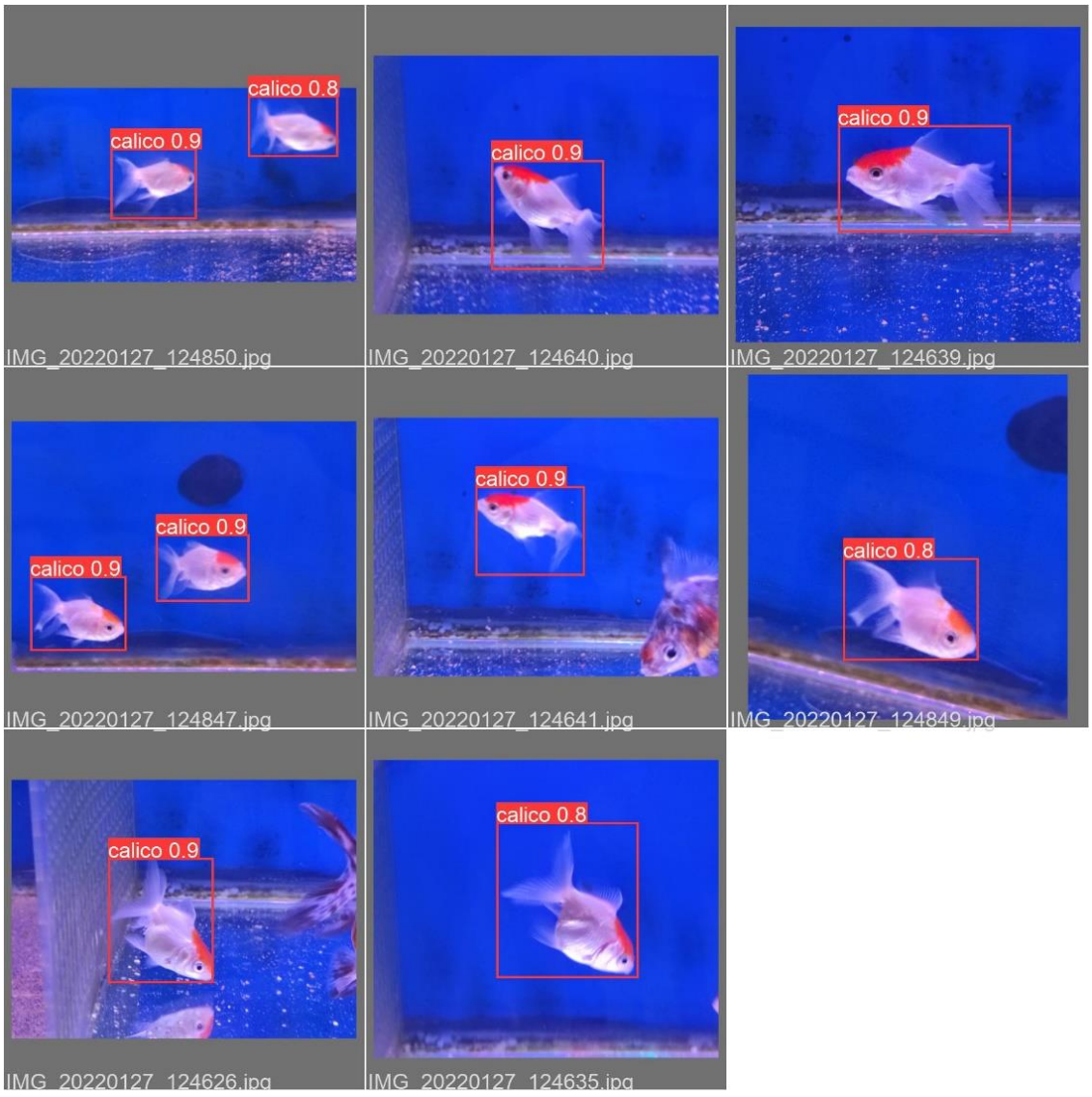


Figure 4-3-4-2 Sample Detection Result Generated By YOLOv5 Model

4.3.5 Raspberry Pi

After developing the MobileNet detection model, the model is ready to be embedded into the Raspberry Pi SBC. In this section, the details of the Raspberry Pi would be discussed. Figure 4-3-5-1 shows how the Raspberry Pi setup looks like.

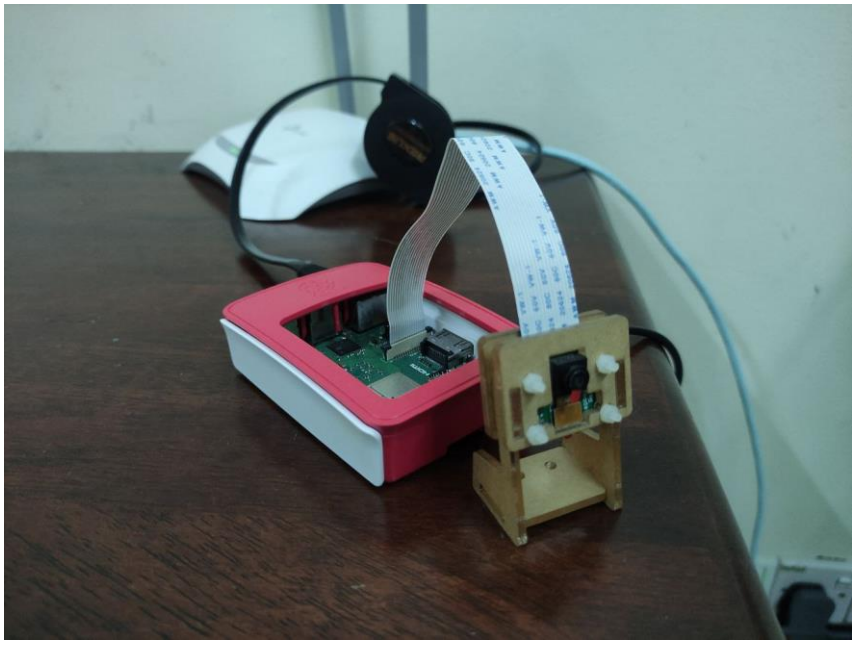


Figure 4-3-5-1 Raspberry Pi Setup

First of all, the SBC does not come with any prebuilt storage to be booted. For that reason, users should prepapre an empty SD card and burn an operating system (OS) image into it using the Raspberry Pi OS imager. For that matter, the OS image that would be used is the "Raspberry Pi OS (Legacy) with desktop". Before burning the OS image, it is important to enable the Secure Shell (SSH) because the setup does not involves with an external monitor, keyboard, and mouse. In other words, all the configuration would be done using a laptop that is connected to the SBC via SSH. After that, the SD card is ready to be inserted into the SBC booted by the SBC. Next, the camera module should be attached to the SBC. Also, since the SSH requires internet connection, the SBC should be connected to a router or access point using an ethernet cable. Once it is finished, the SBC is ready to power up using the power adapter as described in section 4.2. After powering up the SBC, all of the software configuration is done via SSH connection and during the process, the VNC Viewer services is enabled and the Tensorflow library has also been installed in order to run the deep learning model. Also, VNC Viewer is a service that allow users to gain remote access to a particular device with graphical user interface and this allow users to be able to monitor the fishes from a different device that is accessing

the SBC remotely. Once all the setup is completed, the detection model is tested and Figure 4-3-5-2 shows the sample output of the detection.

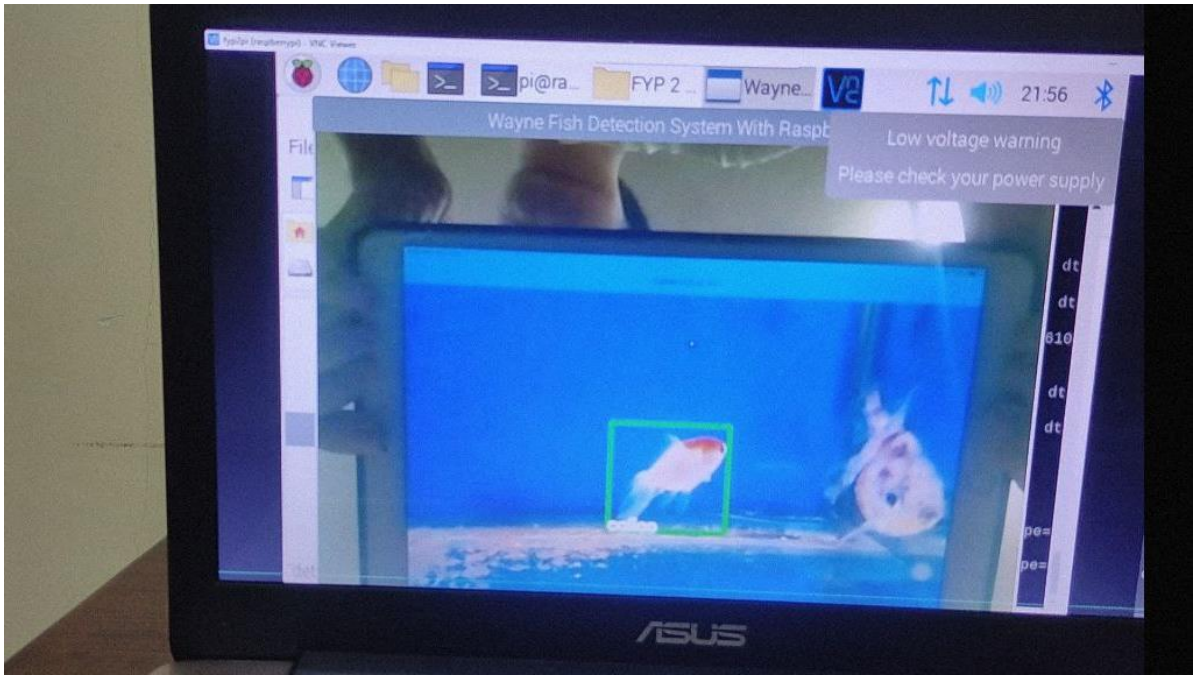


Figure 4-3-5-2 Sample Detection Output

4.3.6 Arduino

In this project, it is mentioned that Arduino would be used as a supportive element to improve the usage of the overall fish monitoring system and Figure 4-3-6-1 shows the design of the Arduino circuit board. Based on the figure, it consists of 2 sensors which is the temperature sensor and the LDR light sensor. If the LDR sensor senses the environment is dark, the green LED at the bottom left would be triggered to indicate the environment needs more lighting. As for the temperature sensor, it would trigger each of the three LEDs in the middle according to the condition. In this case, the Arduino is designed to trigger the red LED when the temperature surrounding is reaching 27°C or higher and the yellow LED would be triggered when the temperature is lower than 24°C. Otherwise, the green LED would be triggered to indicate the temperature is in between 24°C and 27°C which is good. To understand the details of the coding implementation, the source code has been documented in the appendices section.

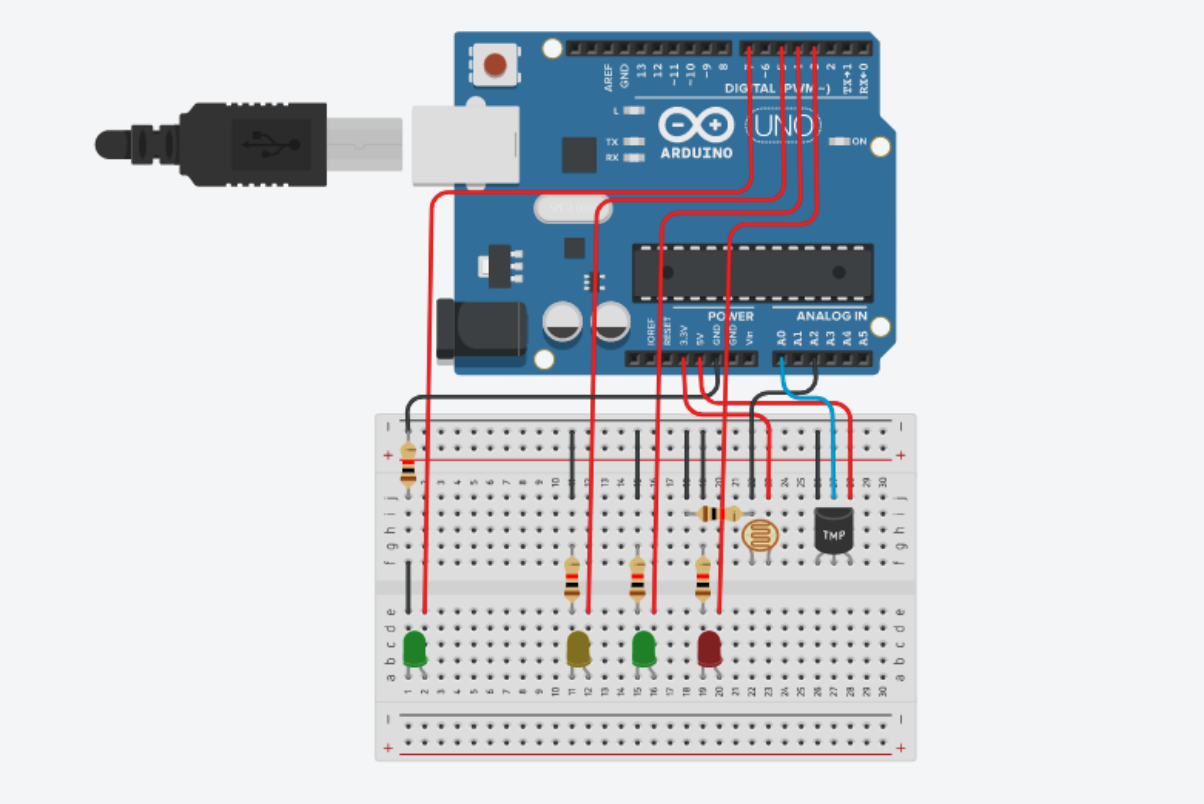


Figure 4-3-6-1 Arduino Circuit Design

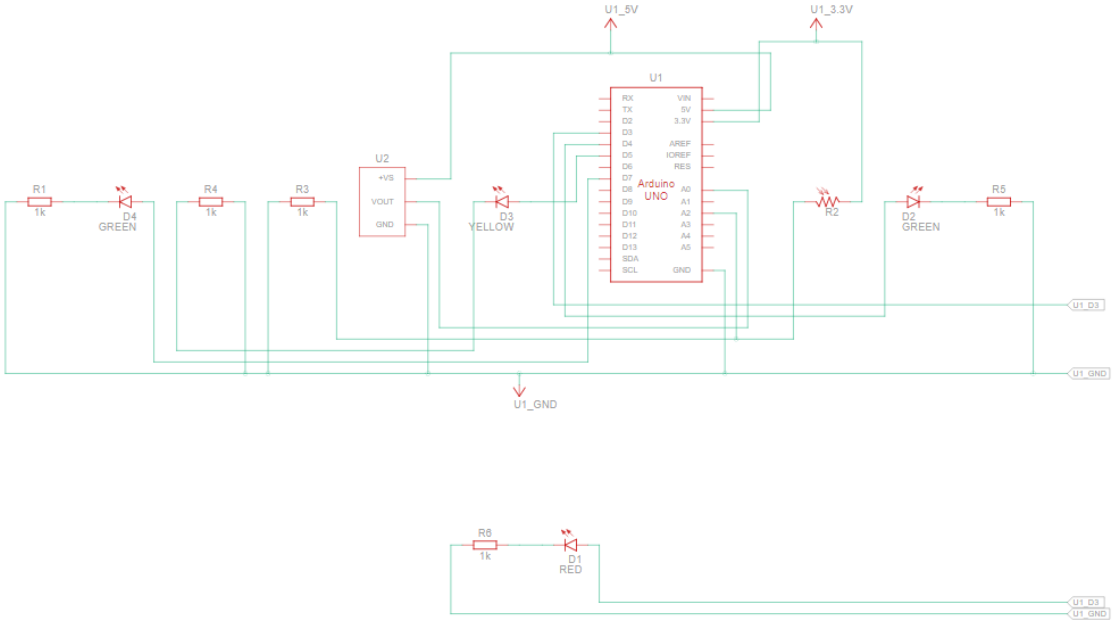


Figure 4-3-6-2 Schematic View of Figure 4-3-6-1

4.4 System Components Interaction Operations

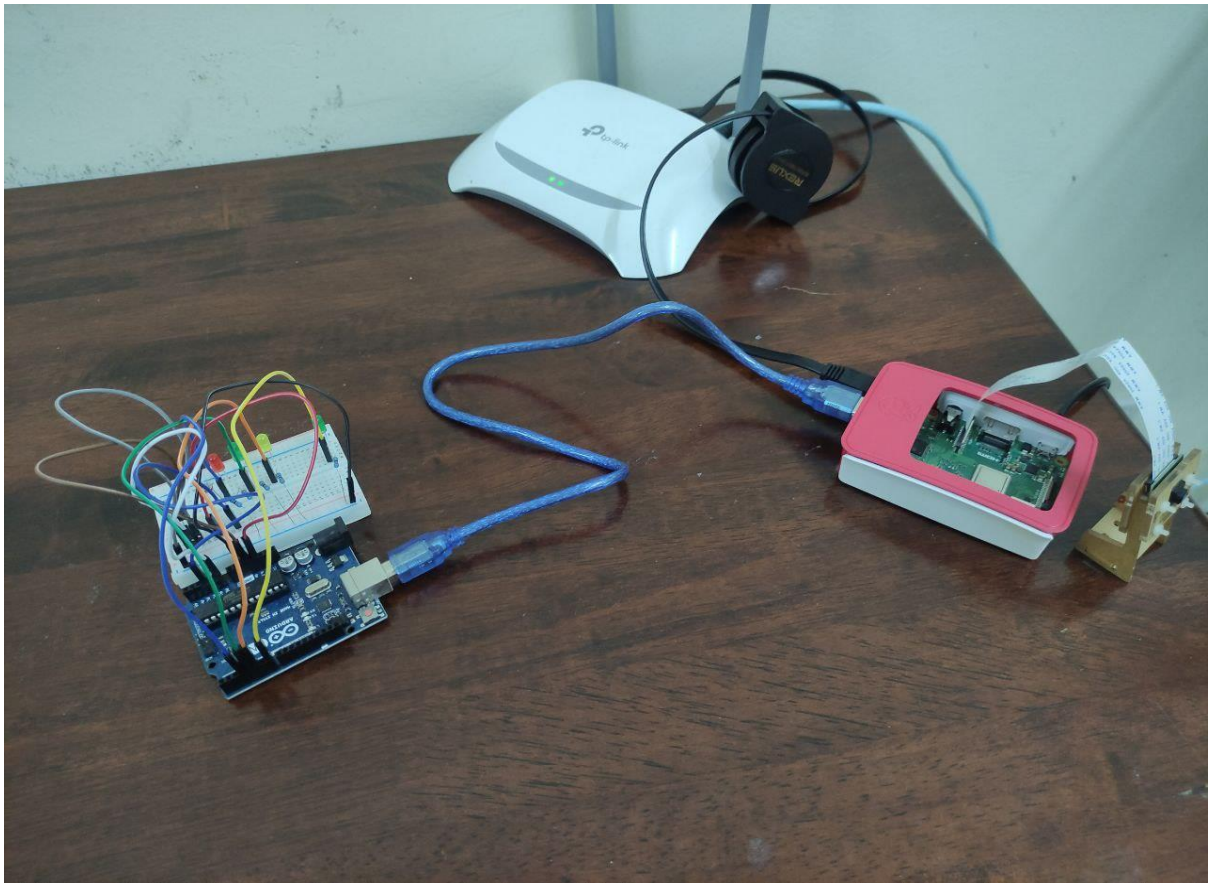


Figure 4-4-1 Whole System Setup

After completing the all of the necessary setup as discussed from section 4.3, the final deliverable of the whole project is as shown in Figure 4-4-1. Besides, based on the figure above, the Arduino is connected to the Raspberry Pi using a blue USB A-to-B cable in order to receive the power supply from the Raspberry Pi. Further, the Arduino would start to function as long as it has received the power supply directly from the Raspberry Pi. As for the Raspberry Pi, the SBC itself is powered by a 5V/3A Raspberry Pi 3 power adapter. Further, the SBC is also connected to a router using an ethernet cable in order to allow SSH and VNC communication for remote access purposes. With all those setup, the whole system is ready to be used. First of all, user may use the detection services by using a device such as a laptop to gain access to the Raspberry Pi remotely using the VNC service according to the IP address of the Raspberry Pi in the network that the SBC is connected to. While users are connecting to the SBC, they would be requested to provide the correct credential such as username and password to gain access to the SBC. Once it is all connected, users are allowed to proceed to running the detection program to monitor the fishes and the sample output of the detection could be seen in Figure 4-3-5-2.

CHAPTER 5

System Simulation

Based on the previous discussion, the necessary information required to build the whole system has been explained. Therefore, the following section would show the system simulation.

5.1 Hardware Setup

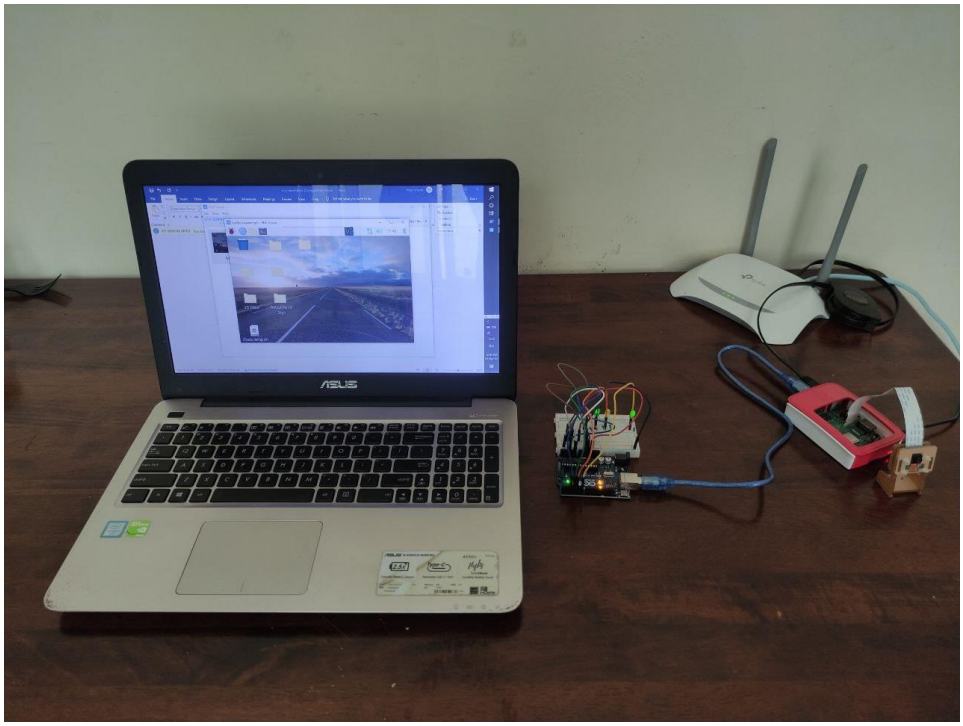


Figure 5-1-1 Hardware Setup

Figure 5-1-1 shows the setup required for the hardware. Based on the figure, the Raspberry Pi is connected to a power supply and a router in order to be able to communicate to the laptop as shown via VNC service. Then, the Arduino could be connected directly to the Raspberry Pi and it would start the preconfigured program. In addition, laptop would be a great device to be used to take control over the Raspberry Pi since it is able to connect to the router wirelessly. Any devices such as a tablet or smartphone is applicable to substitute the laptop as long as the device supports VNC service.

5.2 Software Setup

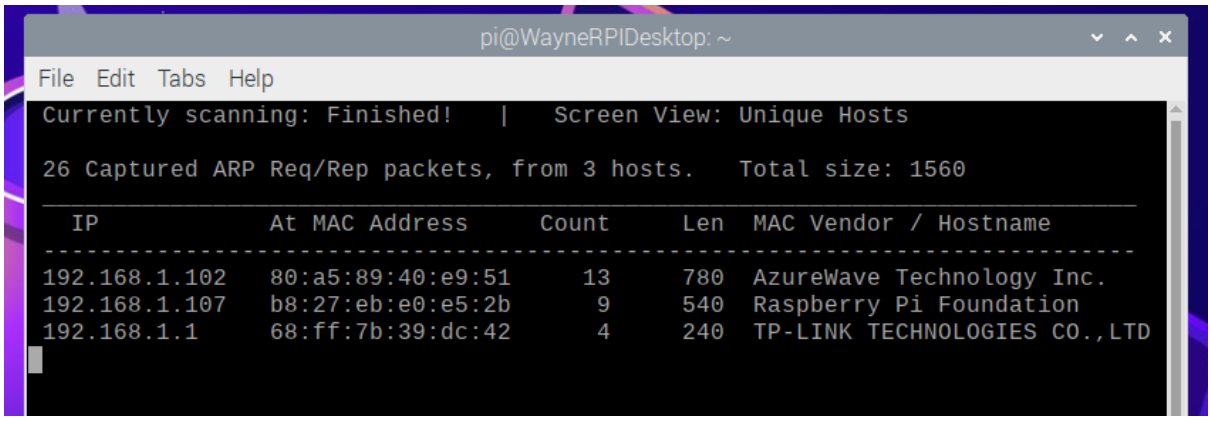


Figure 5-2-1 netdiscover

Further, the Raspberry Pi is now powered up and connected to a router via ethernet cable. After that, the Raspberry Pi should now have internet connection with an IP address assigned to it. To connect the laptop to the Raspberry Pi, it is important to identify the IP address of the Raspberry Pi that has been assigned to it because VNC viewer connects a device to another based on the IP address of the device that is going to be connected. To identify the IP address of the Raspberry Pi, one of the easiest way is to access to the router and check the IP address assigned to the SBC. Besides, if the router is not accessible due to permission issue, it is still possible to reveal the IP address of the SBC using an utility available in most Linux distribution called “netdiscover”. During the setup process, a dummy Raspberry Pi OS virtual machine has been made using VirtualBox in order to use “netdiscover” to reveal the IP address of the SBC and the sample output of the command is shown in Figure 5-2-1. Based on the figure, the IP address of the Raspberry Pi SBC is 192.168.1.107. Once the IP address is retrieved, the user could access to the Raspberry Pi using the VNC viewer according to the IP address as mentioned earlier. Once the connection is established, the VNC viewer should produce the similar output as shown in Figure 5-2-2.

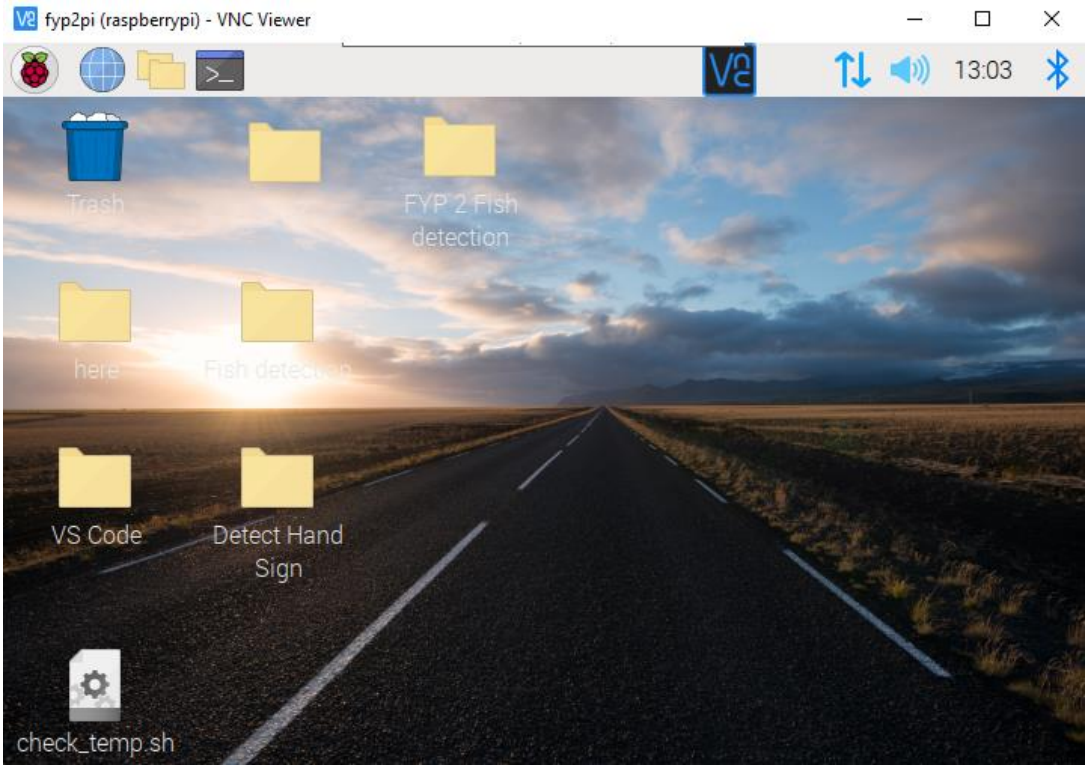


Figure 5-2-2 VNC Viewer

5.3 System Operation

The following shows the sample output of the detection service while using the system from a laptop's perspective and the sample output of the Arduino.

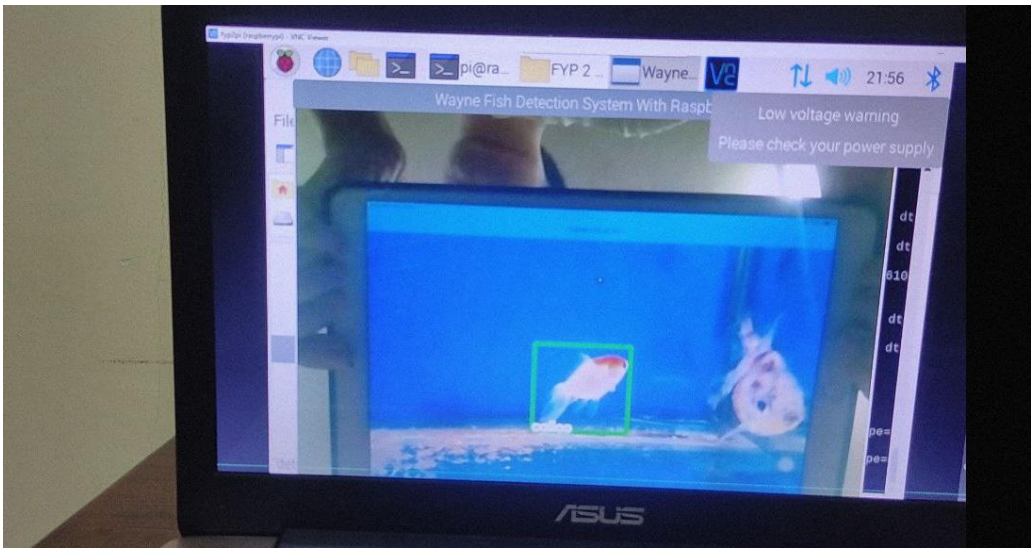


Figure 5-3-1 Sample Detection Output

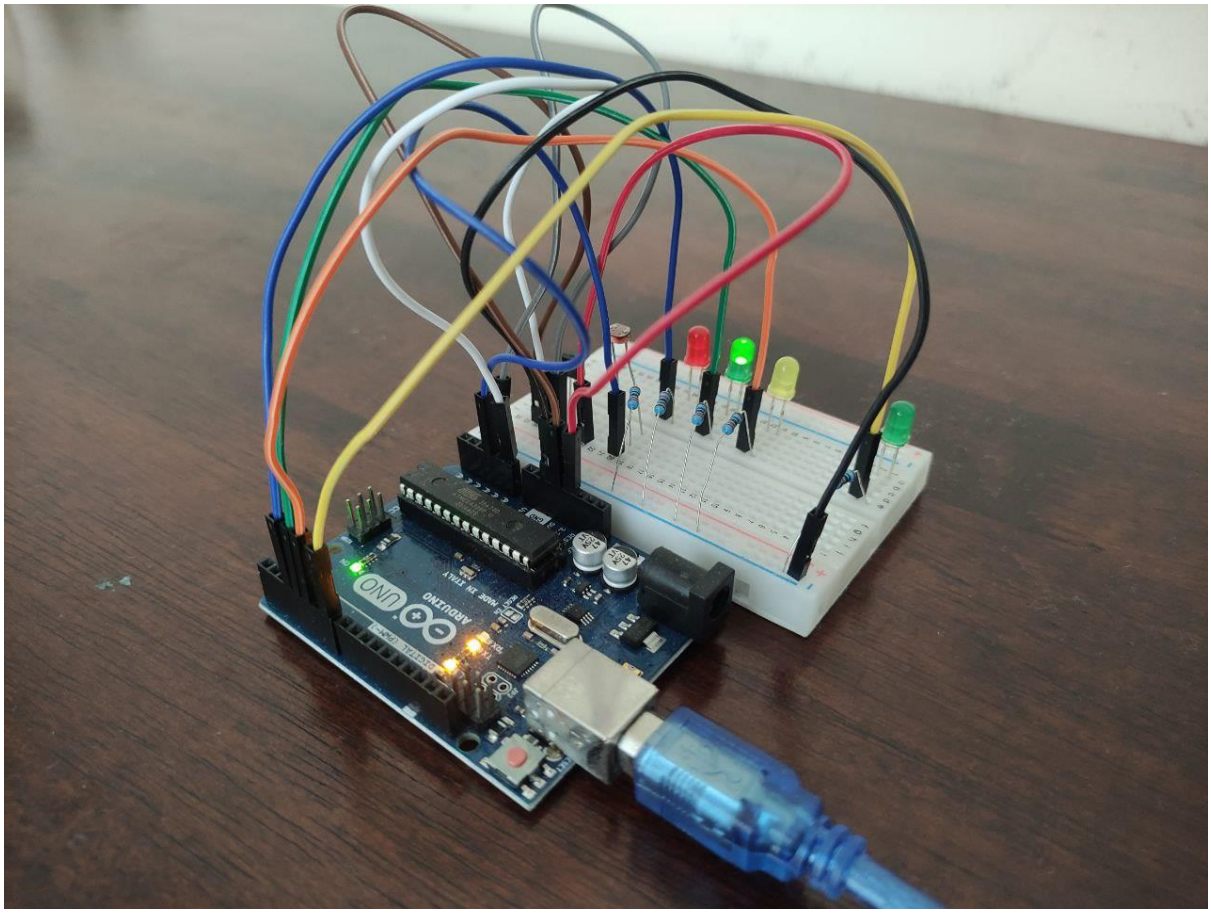


Figure 5-3-2 Sample Output of Arduino

5.4 Concluding Remark

In short, this chapter has explained the necessary setup for the hardware and software to use the whole system. In the hardware setup, the Raspberry Pi should be connected to a power supply and a router via ethernet cable for internet access purposes. After that, the Arduino could be connected to the Raspberry Pi directly. To use the system, any devices that support VNC viewer could access to the SBC based on its IP address. Once the device is connected to the SBC, the detection system is ready to be used and the Arduino would start to perform its tasks right after it is connected to the Raspberry Pi.

CHAPTER 6

System Evaluation And Discussion

6.1 System Testing and Performance Metrics

After the system implementation, the system is prepared for testing and evaluation purposes. Besides, due to the fact of Covid-19 pandemic, the movement restriction is very high and limiting availability of many location for testing the system. Because of that, the system was tested and evaluated in a room with internet connection. As for the details, the disucssion has been made in chapter 5, section 5.1. Further, based on the result of the SSD MobileNet V2 FPNLite 320x320 in Table 4-3-3-3, the model was able to perform the detection services with the accuracy result of 82.43% and recall result of 67%. Compared to the model developed in the previous project in FYP 1, the accuracy result of the model developed in this project is over three times better in terms of detection accuracy. Meanwhile, the recall metrics obtained in this project is almost the double of the performance of the trained model obtained from FYP 1. Also, the Arduino also plays an important role to detect the surrounding temperature and the environment's lighting, and it would trigger specific LED based on the data received from the sensors. Therefore, to complete the test, the following is the list of verification steps that have been identified.

1. Test if the Raspberry Pi could perform the detection services.
2. Test the temperature sensor controlled by the Arduino.
3. Test the LDR light sensor controlled by the Arduino.

6.2 Testing Setup and Result

6.2.1 Result of Raspberry Pi's Detection Services

To test the detection services, the Raspberry Pi is first connected to a router via ethernet connection for supporting the VNC Viewer communication and this is to allow any devices that are under the same network to be able to control over the Raspberry Pi. Then, after running the detection script, the camera attached starts to capture the images and pass the image to the detection model. During the test, it is found that the model was not always able to generate the desire output in terms of accuracy. For example, Figure 6-2-1-1 shows a sample of a true positive fish detection output, Figure 6-2-1-2 shows a sample of false positive fish detection

output and Figure 6-2-1-3 shows a sample of false negative fish detection output. Besides, even though the Raspberry Pi is able to perform the detection services, the process is still not efficient enough to be considered as real time due to the limited compute resources available from Raspberry Pi. In other words, the detection process is not fast compared to many existing system that is implemented on a more powerful machine.

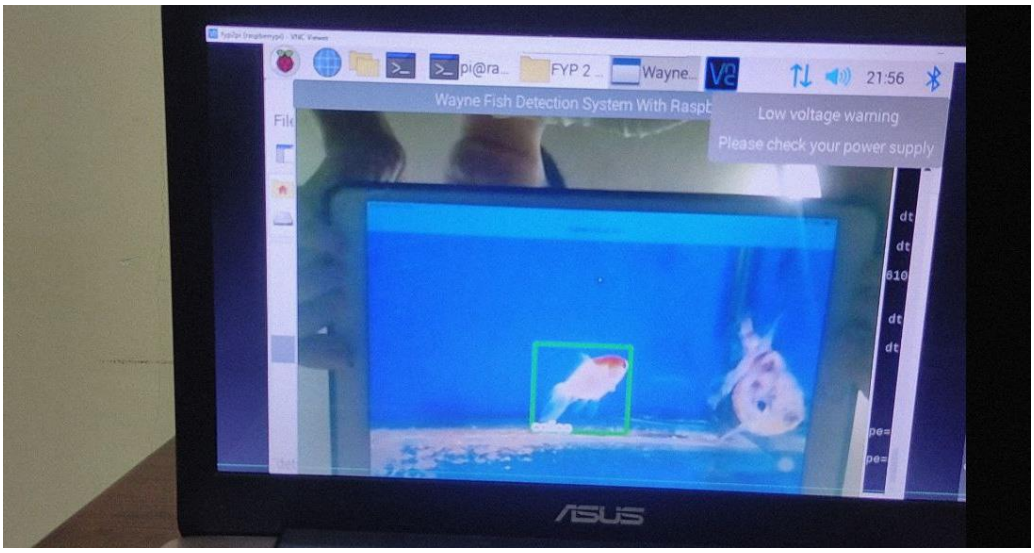


Figure 6-2-1-1 True Positive Detection Output

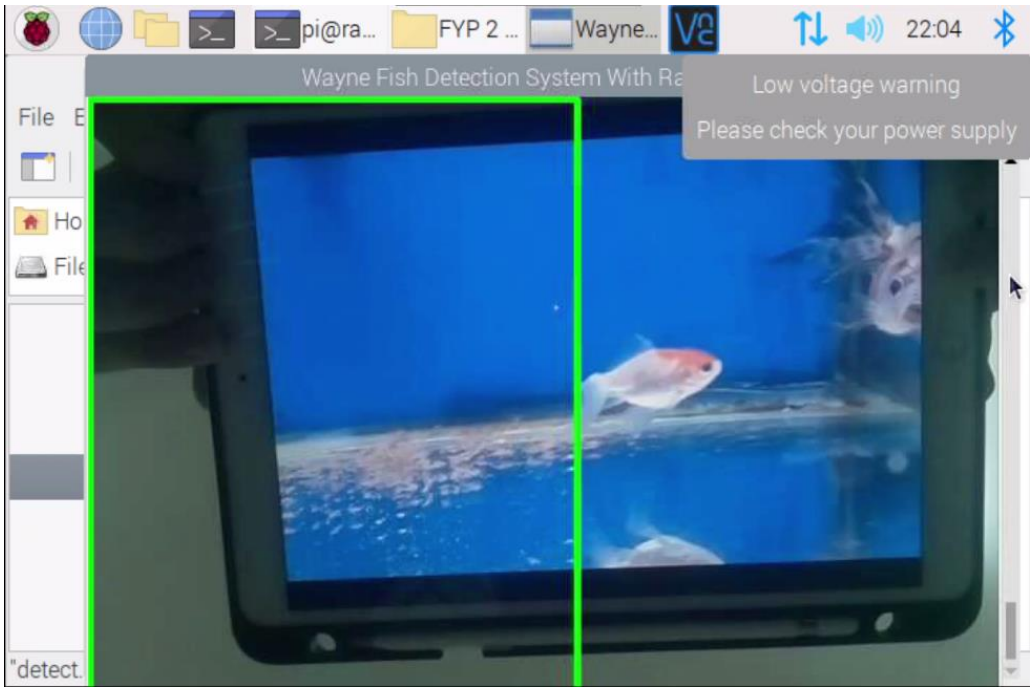


Figure 6-2-1-2 False Positive Detection Output

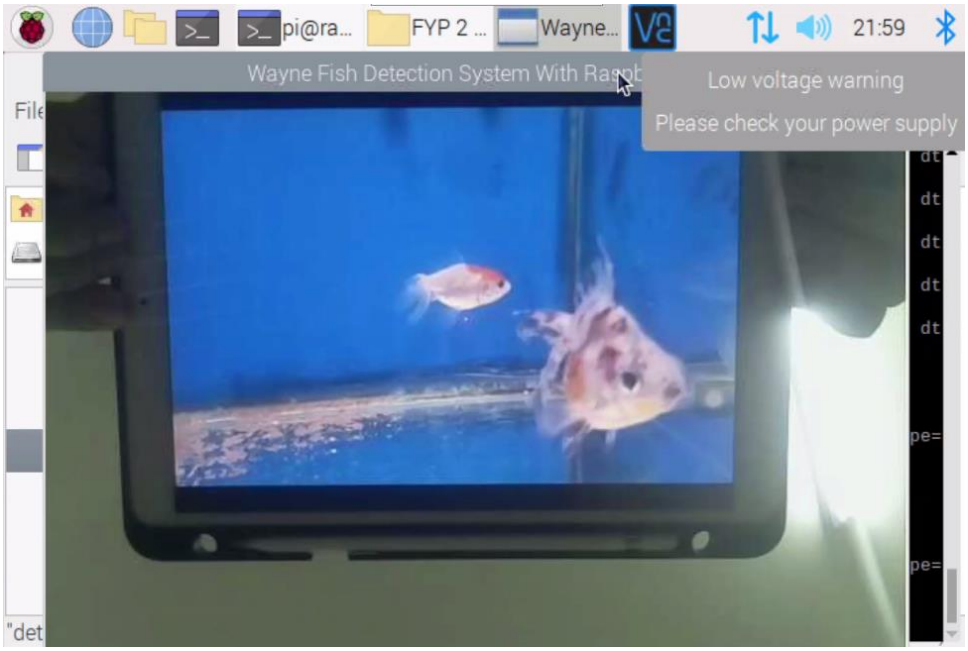


Figure 6-2-1-3 False Negative Detection Output

6.2.2 Result of Temperature Monitoring Module

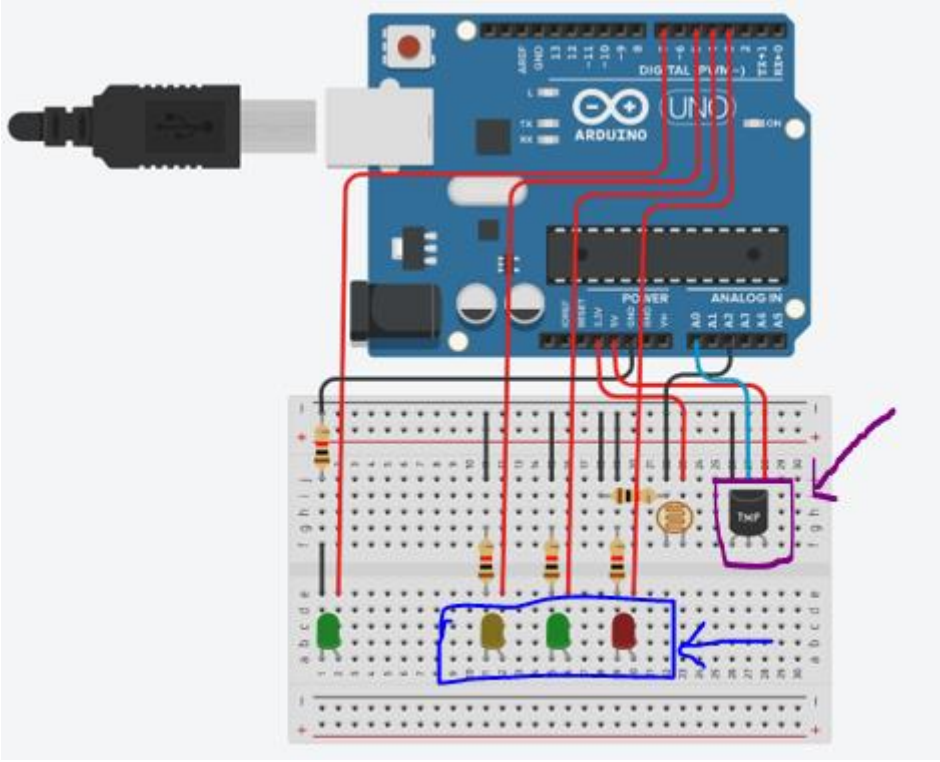


Figure 6-2-2-1 Temperature Sensor and LED Indicator

Based on the diagram as shown in Figure 6-2-2-1, it is the design of the electrical circuit that would be interact with the Arduino board. In this case, it is important to understand which part of the circuit is responsible for sensoring the surrounding temperature and trigger specific

component to indicate the temperature status. Based on the diagram, the component highlighted in purple is the temperature sensor and the LEDs highlighted in blue would be used to indicate the status of the environment temperature. As discussed, the red LED is triggered when the temperature is 27°C or above to indicate the temperature is very high. Besides, the yellow LED would be triggered whenever the temperature is lower than 24°C to indicate that the temperature is lower than the optimum temperature. If the temperature is in between 24°C and 27°C, the green LED in the middle of the red and yellow LED would be triggered.

To test the temperature monitoring module, the Arduino would be placed in three different location with different environment temperature. The first location being used to test the module is inside a normal room since it is the easiest location that has the temperature in between 24°C and 27°C. The result of the Arduino is as shown in Figure 6-2-2-2. Next, to trigger the red LED, some water is boiled to create some steam vapour. When the water is at its boiled, the Arduino is placed where the steam vapours and the result is shown in Figure 6-2-2-3. Lastly, To trigger the yellow LED, the experiment is done just by simply putting the Arduino inside refrigerator and the result is shown in Figure 6-2-2-4.

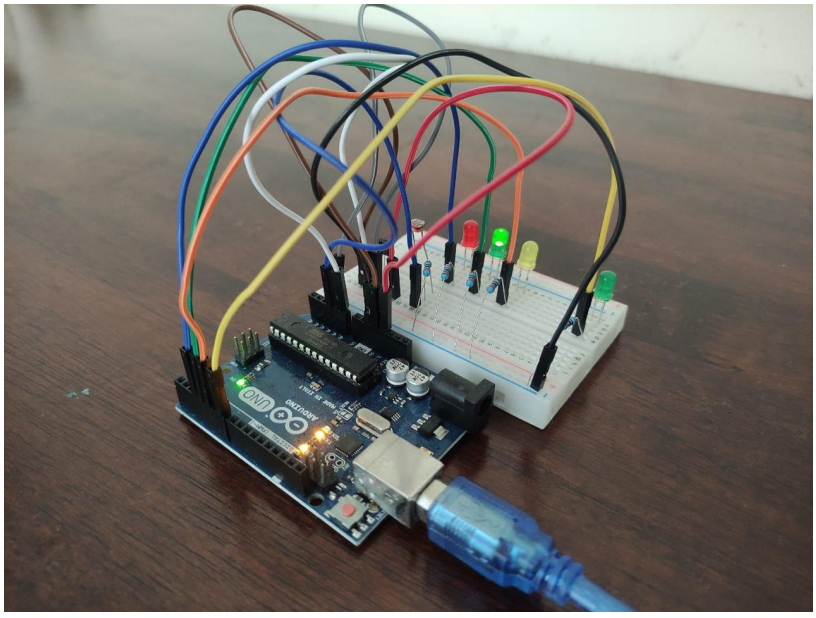


Figure 6-2-2-2 Normal Temperature

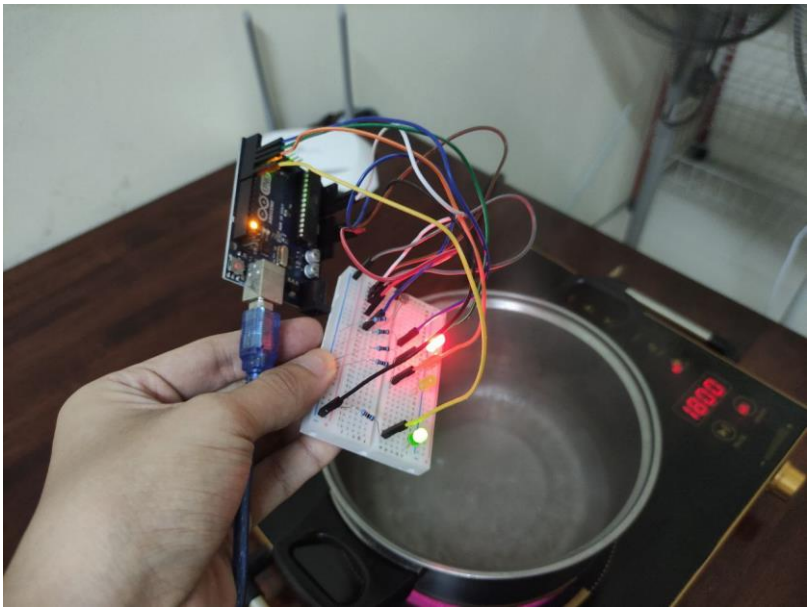


Figure 6-2-2-3 Hot Temperature

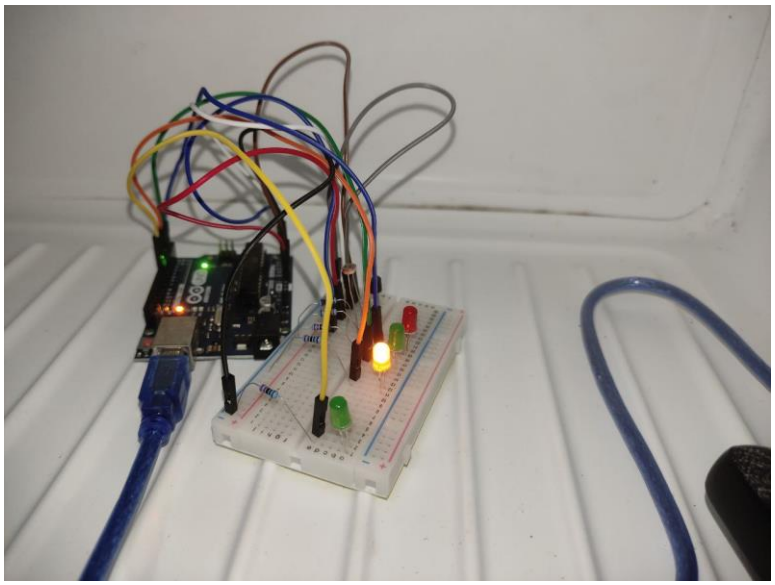


Figure 6-2-2-4 Cold Temperature

6.2.3 Result of Light Level Monitoring Module

In this project, the Arduino has also been used as the component to sense if the light level surrounding it is bright enough to indicate if the Raspberry Pi requires more light to capture the images of the fish clear. Based on the design of the circuit board, Figure 6-2-3-1 has shown the LDR light sensor and the green LED indicator that would be triggered based on certain sitaution. For example, if the environment is dark, the LED would be triggered to indicate that more light is required to make the environment brighter to allow the Raspberry Pi to capture a clearer image.

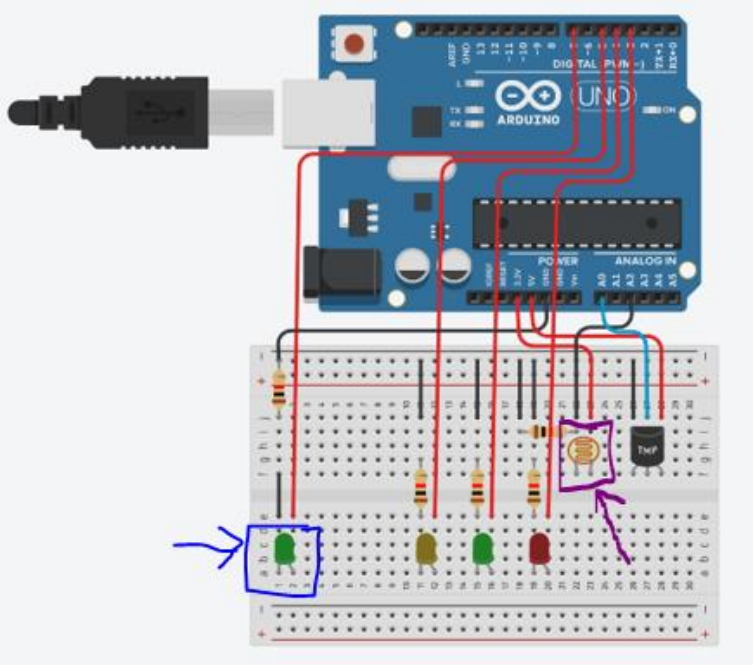


Figure 6-2-3-1 LDR Light Sensor and LED Indicator (Torch LED)

To test module, the Arduino is placed in two different rooms. One of the rooms as mentioned is a dark room and the other one is a bright room. In Figure 6-2-3-2, the LED indicator is triggered in a dark room and representing that the room needs more light and Figure 6-2-3-3 has shown that the LED indicator is not being triggered and indicates that the room has sufficient light for the Raspberry Pi to capture a clear image.

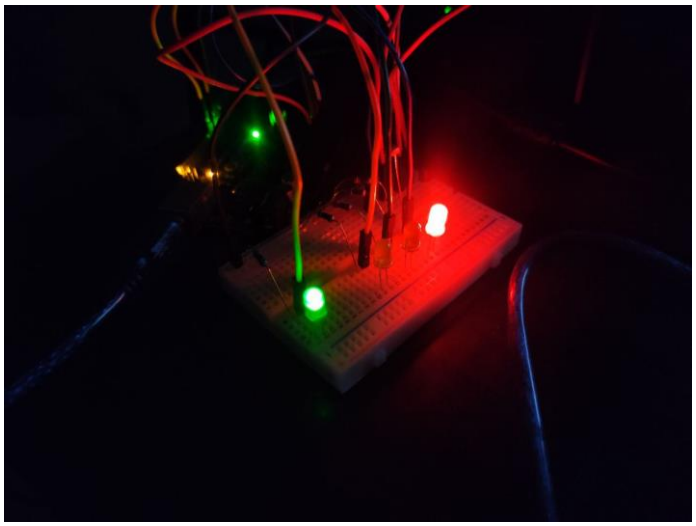


Figure 6-2-3-2 Arduino In a Dark Room

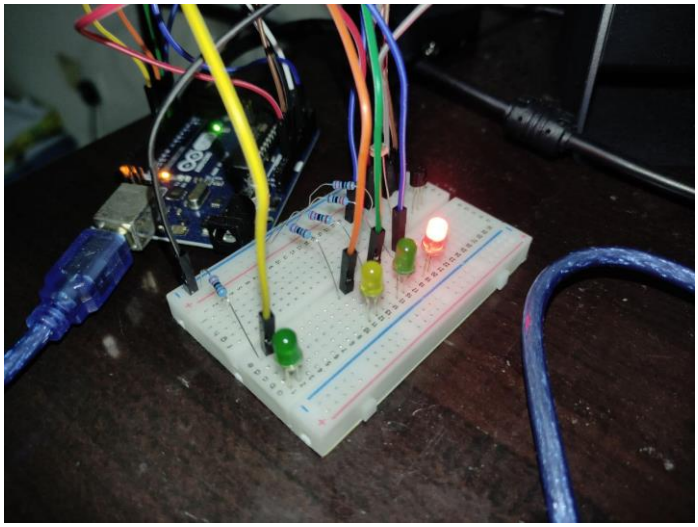


Figure 6-2-3-3 Arduino In a Bright Room

6.3 Project Challenges

In this project, there are a lot of studies and experiments being accomplished and carried in order to complete this project. In fact, the journey of completing this project was not very smooth due to various challenges faced during the studies and development of this project. First of all, this project involves studies from different field such as machine learning, image processing, eletrical components, electrical circuit design, networking and more. Besides, according to personal degree course, the main studies is focusing on IT development. In this case, since the project involves electrical circuit design, extra effort is required to study the fundamental of the eletrical components and circuit design. In electrical circuit design, it is possible to cause incidents such as fire and even electric shock if no proper preparation or practices is made. In short, the first challenge that is faced during the project development is that the extra effort required for studying the related and required field to complete the project. Next, to train the detection model for project, the image dataset that has been used was collected manually using a smartphone and the reason that this approach is used instead of using the existing image dataset from various popular databases such as ImageNet is because of limited access to those databases. For that reason, this has made the project development duration longer and the image dataset prepared may not be as high quality as the image dataset from those databases for creating a high accuracy deep learning model. Furthermore, there is also a challenge faced during the project development and that is the limited compute power of the Raspberry Pi. In this project, it was expected to deliver a detection system that could be used in real time. However, due to the design of the Raspberry Pi SBC, the hardware is not able to

function efficient enough for real time monitoring. Initially, it was planned to obtain a Google Coral USB accelerator to boost up the speed of the process of detection system up to 3 times of its original speed. But, the availability of the item is very limited and the cost is also very high as well. Therefore, the proposed solution may not perform as effective and efficient as the existing systems used in industries.

6.4 Objectives Evaluation

According to the project objectives, there are total of three objectives being defined as the end point of this project. First of all, one of the objectives as stated is to develop a fish monitoring system that provides visual monitoring services by developing a fish detection system. In this project, the objective as stated earlier has been achieved by studying and developing a deep learning module based on the architecture named MobileNetV2 which is famous for its light weighted and highly accurate recognition performance. Second of all, another objective of this project it to develop a fish monitoring system that allow users to safely monitor the fishes visually without the need to stay at the farm location. To achieve the objective, a simple approach has been used which is by implementing the remote access services called VNC Viewer available in the Raspberry Pi SBC. By implementing the services, user could use their device to remotely access the Raspberry Pi SBC to use the system as long as they are under the same network or subnet. Last but not least, the final objective that is being defined in this project is to develop an improved version of the FYP 1 based monitoring system in terms of AI detection accuracy and functionality. In this project, a different approach to prepare the image dataset has been deployed for training the deep learning module and has achieved over 80% of detection accuracy. Besides, to improve the overall functionality, two different subsystem has been designed and added into the proposed system which is the temperature and light level monitoring module. With that, the quality of the monitoring process could be enhanced to keep the aquatic culture from many risks such as illness. In short, all of the objectives stated in this project have been achieved successfully.

6.5 Concluding Remark

In this chapter, several experiments have been carried in order to evaluate the overall performance of the fish monitoring system. Besides, to evaluate the system, several tests have been designed for experimenting the performance and the functionality of the proposed system. After completing the experiment, the result shows that the deep learning module was able to

perform the detection services with satisfied performance. Besides, the temperature and light level monitoring submodules have also been tested with several test cases and the result generated was better than expected. While completing this project, there are many challenges were faced that had been affecting the quality of the project deliverable in many ways. However, even though there were many problems encountered during the project development, the final deliverable still has fulfilled the objectives as defined in this project. Therefore, the project is still considered as a successful project.

CHAPTER 7

Conclusion And Recommendation

7.1 Conclusion

As the conclusion, aquaculture farm is a very important element in the industries as the technologies evolve for various reasons such as species cultivation and food production. Besides, to preserve or improve the farm, it is important to have a well planned monitoring process to keep the farm running. For that reason, studies in machine learning, image processing and the foundation in electrical circuit have been made in order to develop a fish monitoring system for improving the existing fish monitoring process. During the development process, image dataset of specific type of fish has been prepared for training the deep learning module. Speaking of deep learning module, the architecture of the model is developed based on MobileNetV2 and the detection accuracy has reached up to 82.43% and recall metrics of 67%. For study purposes, a YOLOv5 based deep learning model has been developed and trained with the same image dataset and training configuration because some studies have proven that YOLOv5 performs better than MobileNetV2 in terms of detection accuracy, recall and even speed. Once the model is developed, evaluation process is made and found that the YOLOv5 model was able to achieve 86.14% in detection accuracy and 100% in recall metrics. However, due to the limited time and personal capability, the YOLOv5 model was not being converted into a suitable file format for comparing the detection speed of both models and was not being deployed in the Raspberry Pi as the main model to provide detection services. After that, the submodule of temperature and light level monitoring have also been implemented to further improve the overall fish monitoring process. To implement the subsystems, an Arduino microcontroller is used to control the sensors and trigger specific LED to indicate the environment status. At the end of the project, the proposed system was able to generate the expected and satisfactory result, hence the objectives of the project are achieved.

7.2 Recommendation

Based on the final deliverable of the project, there are still many aspects that could be improved to enhance the overall quality of the performance of the proposed system. First and foremost, it is mentioned that the SBC to be used to run the deep learning model is the

Raspberry Pi 3B+. According to the Raspberry Pi documentation, the processor installed in the SBC is not as strong as many modern processors such as Intel and AMD Ryzen based processor. Because of that, the performance of the detection services is not efficient since the deep learning module is a CPU intensive process. To improve the speed of the detection system, it is suggested to use a better SBC such as UDOO Bolt SBC that has a built in AMD Ryzen processor. Other than using a stronger SBC, the speed of the Raspberry Pi still could be boosted by using a Google Coral USB accelerator. Based on the reviews of the USB accelerator, it could improve the detection speed up to over double of the original performance of a Raspberry Pi. Besides, based on the result of the detection system, the model does not always generate very accurate detection even though the detection accuracy was documented at over 80%. The reason such case happened in this project is because of image dataset prepared during the development process was not sufficient enough to train the deep learning module to generalize the object well. In other words, it is highly recommend to use a better image dataset available from popular databases such as ImageNet as some of the researchers have proven that the different dataset could impact the accuracy performance of a deep learning model. Furthermore, the Raspberry Pi configured in this project is designed to connect to the internet via ethernet for remote access purposes. For better mobility, it is still possible to allow the Raspberry Pi to connect to the internet wirelessly. In other words, if the Raspberry Pi is planned to be used with higher mobility action, it is recommended to configure the SBC to be connected to a router or an access point via Wi-Fi. Last but not least, there are a few aspects that could be improved in the Arduino submodule. As mentioned, the Arduino controller would be used to control and run the temperature and light level monitoring system. In fact, the Arduino microcontroller is able to control and trigger more than just some LEDs and even the temperature and light sensor. Due to the limited time and monetary resources, the Arduino is used as a demonstration to show that the component itself could control many different components and take an action based on the condition being programmed in the microcontroller. In this project, the LM35 temperature sensor is not very suitable to be used to measure the temperature of the water in the aquaculture farm since the component itself is not waterproof. For that reason, the LM35 temperature sensor could be replaced with a better waterproof sensor to allow the system to calculate the temperature of the water more precisely. Besides, it is stated that a LED named "Torch LED" is used to indicate the illuminance of the environment or to inform users if more light is required to make the environment brighter for better visual monitoring process. Initially, instead of using a small LED, it was planned to implement the LDR sensor to provide the

illuminance data of the environment to the Arduino to allow the microcontroller to turn on some brighter bulbs to provide light to the aquaculture farm. With this, the users would not need to manually turn on or off the light during the monitoring process. The reason that such implementation is not made is that this implementation would require more electronic components especially a relay since turning on a bulb would require a large amount of power. Therefore, it is still highly recommended to use a better sensors as mentioned earlier to enhance the overall system functionality.

REFERENCES

[1] World Food Program USA, "Nearly 60% of the World's Hungriest People Live in Just a Few Countries. Why?", wfpusa.org. https://www.wfpusa.org/articles/60-percent-of-the-worlds-hungry-live-in-just-8-countries-why/ (accessed Jan. 31 2022).

[2] BAE SYSTEMS, "What are single-board computers?", baesystems.com. https://www.baesystems.com/en-us/definition/what-are-single-board-computers (acceseed Feb. 28 2022).

[3] Long. M, "Best Single-Board Computers for Every Use and Budget in 2021", electromaker.io. https://www.electromaker.io/blog/article/best-single-board-computers (accessed Mar. 1 2022).

[4] Raspberry Pi, "Raspberry Pi 3 B+ Power Supply", forums.raspberrypi.com. https://forums.raspberrypi.com/viewtopic.php?t=228020 (accessed Mar. 1 2022).

[5] Chacos. B, Chiappetta. M, "How to choose the best PC power supply", pcworld.com. https://www.pcworld.com/article/456490/how-to-pick-the-best-pc-power-supply.html (accessed Mar. 1 2022).

[6] Upton. E, "Raspberry Pi 3 Model B+ on sale now at $35", raspberrypi.com. https://www.raspberrypi.com/news/raspberry-pi-3-model-bplus-sale-now-35/ (accessed Mar. 1 2022).

[7] Linus Tech Tips, *10 ways Linux is just better!*. Feb. 25, 2020. Accessed on: Mar. 2, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=mAFMJ1LnQu8

[8] Raspberry Pi, "Operating system images", raspberrypi.com. https://www.raspberrypi.com/software/operating-systems/#raspberry-pi-os-32-bit (accessed Mar. 2 2022).

[9] Jeff Geerling, *It's official: Raspberry Pi goes 64-bit*. Feb. 4, 2022. Accessed on: Jan. 20, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=ET-DJLnNX-Q&t=71s

[10] Keim. R, "What Is a Microcontroller? The Defining Characteristics and Architecture of a Common Component", allaboutcircuits.com. https://www.allaboutcircuits.com/technical-articles/what-is-a-microcontroller-introduction-component-characteristics-component/ (accessed Mar. 2 2022).

[11] ARDUINO EDUCATION, “EMPOWER SCIENTISTS AND ARTISTS OF THE FUTURE”, arduino.cc. https://www.arduino.cc/education (accessed Mar. 3 2022).

[12] Circuito Team, “EVERYTHING YOU NEED TO KNOW ABOUT ARDUINO CODE”, circuito.io. https://www.circuito.io/blog/arduino-code/#:~:text=Arduino%20code%20is%20written%20in,and%20compiled%20to%20machine%20language. (accessed Mar. 2 2022).

[13] Mike. W, “Why Python is Good for Machine Learning”, section.io. https://www.section.io/engineering-education/why-python-is-good-for-machine-learning/#:~:text=Simplicity%3A%20Python%20code%20is%20concise,compared%20to%20many%20programming%20languages. (accessed Mar. 3 2022).

[14] Brownlee. J, “What is Deep Learning?”, machinelearningmastery.com. https://machinelearningmastery.com/what-is-deep-learning/ (accessed Mar. 3 2022).

[15] Gurucharan. M, “Basic CNN Architecture: Explaining 5 Layers of Convolutional Neural Network”, upgrad.com. https://www.upgrad.com/blog/basic-cnn-architecture/ (accessed Mar. 4 2022).

[16] Saha. S, “A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way”, towardsdatascience.com. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53 (accessed Jan. 20 2022).

[17] Hariharan. B, Girshick. R, “Low-shot Visual Recognition by Shrinking and Hallucinating Features”, in *2017 IEEE International Conference on Computer Vision*, pp. 3037-3046, 2017.

[18] Jalal. A, Salman. A, Mian. A, Shortis. M, Shafait. F, “Fish detection and species classification in underwater environments using deep learning with temporal information”, in *Ecological Informatics*, vol. 57, pp. 1-13, 2020. Accessed on: Mar. 3, 2022. [Online]. Available doi: 101088

[19] Chang. YH, Chung. PL, Lin. HW, “Deep Learning for Object Identification in ROS-based Mobile Robots”, in *Proceedings of IEEE International Conference on Applied System Innovation 2018 IEEE ICASI 2018- Meen, Prior & Lam (Eds)*, pp. 66-69, 2021.

[20] Curtin. BH, Matthews. SJ, “Deep Learning for Inexpensice Image Classification of Wildlife on the Raspberry Pi”, in *IEEE Xplore*.

[21] Iyer. R. V et al, “Comparison of YOLOv3, YOLOv5s and MobileNet-SSD V2 for Real-Time Mask Detection”, irjet.net. https://www.irjet.net/archives/V8/i7/IRJET-V8I7193.pdf (accessed Mar. 10 2022).

[22] Population Reference Bureau, “How Many People Have Ever Lived on Earth?”, prb.org. https://www.prb.org/articles/how-many-people-have-ever-lived-on-earth/ (accessed Feb. 10 2022).

[23] Kidskonnect, “Farm Facts & Worksheets”, kidskonnect.com. https://kidskonnect.com/science/farms/ (accessed Feb. 10 2022).

[24] BRISTOL AQUARIUM, “Captive Breeding: What Is It and Why Is It Vital for Conservation?”, bristolaquarium.co.uk. https://www.bristolaquarium.co.uk/blog/news/captive-breeding-what-is-it-and-why-is-it-vital-for-conservation/#:~:text=Captive%20breeding%20is%20the%20process,easing%20pressure%20on%20wild%20populations. (accessed Feb. 8 2022).

[25] Michael. J, “Farms Are Keeping Endangered Species Alive”, fastcompany.com. https://www.fastcompany.com/2679016/farms-are-keeping-endangered-species-alive (accessed Feb. 8 2022).

[26] Hattersley. L, “Raspberry Pi 3B+ Specs and Benchmarks”, magpi.raspberrypi.com. https://magpi.raspberrypi.com/articles/raspberry-pi-3bplus-specs-benchmarks (accessed Mar. 12 2022).

APPENDIX A

```
const int LED_veryHot = 3;
const int LED_normalTemp = 4;
const int LED_veryCold = 5;
const int LED_torch = 7;

const int LM35_temp_sensor = A0;
const int LDR_light_sensor = A2;
float Vout;

void setup() {
  Serial.begin(9600);
  pinMode(LED_veryHot, OUTPUT);
  pinMode(LED_normalTemp, OUTPUT);
  pinMode(LED_veryCold, OUTPUT);
  pinMode(LED_torch, OUTPUT);

  initialization();
}

void loop() {
  delay(1000);
  int temp = round(calculate_temp());

  sense_light();

  if(temp < 24){
    tooCold();
  }
  else if(temp >= 24 && temp <= 27){
    normalTemp();
  }
  else{
    tooHot();
  }
  delay(1000);
}

void initialization() {
  delay(250);
  digitalWrite(LED_veryHot, HIGH);
  delay(250);
  digitalWrite(LED_normalTemp, HIGH);
  delay(250);
  digitalWrite(LED_veryCold, HIGH);
  delay(250);
  digitalWrite(LED_torch, HIGH);
  delay(800);

  digitalWrite(LED_veryHot, LOW);
  digitalWrite(LED_veryCold, LOW);
  digitalWrite(LED_normalTemp, LOW);
  digitalWrite(LED_torch, LOW);
}
```

```
float calculate_temp() {
  //Get reading from LM35
  int reading = analogRead(LM35_temp_sensor);

  //Convert reading to voltage output (unit is mV)
  Vout = reading * (5000 / 1024);

  //Convert voltage output to temp
  float Temp = Vout / 10;

  //Display temperature
  Serial.print("Temperature: ");
  Serial.print(Temp);
  Serial.print("\xC2\xB0");
  Serial.println("C");
  return Temp;
}

void tooCold() {
  digitalWrite(LED_veryHot, LOW);
  digitalWrite(LED_normalTemp, LOW);
  digitalWrite(LED_veryCold, HIGH);
}

void tooHot() {
  digitalWrite(LED_veryHot, HIGH);
  digitalWrite(LED_normalTemp, LOW);
  digitalWrite(LED_veryCold, LOW);
}

void normalTemp() {
  digitalWrite(LED_veryHot, LOW);
  digitalWrite(LED_normalTemp, HIGH);
  digitalWrite(LED_veryCold, LOW);
}

void sense_light() {
  int LDR_status = analogRead(LDR_light_sensor);

  if (LDR_status <= 30){
    digitalWrite(LED_torch, HIGH);
  }
  else {
    digitalWrite(LED_torch, LOW);
  }

  Serial.print("LDR value: ");
  Serial.println(LDR_status);
}
```

Arduino UNO Code Snippet

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:1**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-None

2. WORK TO BE DONE

-Research about improving deep learning accuracy by training model with different dataset.
-Prepare new dataset.
-Research about the fundamental of eletrical circuit.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-Planned to add temperature and light sensor as a submodule to improve the overall monitoring experience.

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:2**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Research about improving deep learning accuracy by training model with different dataset.
-Prepared new dataset.

2. WORK TO BE DONE

-Research about Arduino.
-Research about the fundamental of eletrical circuit.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-Process of dataset preparation was not very smooth since images are taken manually instead of retrieving from ImageNet database.

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:3**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Research about 3 different models of MobileNet architecture.

2. WORK TO BE DONE

-Research about Arduino.
-Research about the fundamental of eletrical circuit.
-Research about YOLOv5.
-Train the 3 different MobileNet models.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-YOLOv5 is suggested that it could perform faster and more accurate compared to MobileNet.

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:4**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

- Trained one of the MobileNet models.

2. WORK TO BE DONE

-Research about Arduino.
-Research about the fundamental of eletrical circuit.
-Research about YOLOv5.
-Train 2 other MobileNet model.

3. PROBLEMS ENCOUNTERED

-Faulty laptop makes the training duration longer than expected.

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:5**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

- Trained the second MobileNet models.

2. WORK TO BE DONE

-Research about Arduino.
-Research about the fundamental of eletrical circuit.
-Research about YOLOv5.
-Train the last MobileNet model.

3. PROBLEMS ENCOUNTERED

-Faulty laptop makes the training duration longer than expected.

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:6**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Trained the last MobileNet models.
-Research about Arduino.
-Research about the fundamental of eletrical circuit.

2. WORK TO BE DONE

-Buy Arduino and other electronic components such as LED.
-Train the last YOLOv5 model.
-Design electrical circuit (submodule of the proposed system).

3. PROBLEMS ENCOUNTERED

-Faulty laptop makes the training duration longer than expected.

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:7**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Trained the YOLOv5 models.
-Buy Arduino and other electronic components such as LED.

2. WORK TO BE DONE

-Design electrical circuit (submodule of the proposed system).

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:8**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Designed electrical circuit (submodule of the proposed system).

2. WORK TO BE DONE

-Report Writing.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:9**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-None

2. WORK TO BE DONE

-Report Writing.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:10**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-None

2. WORK TO BE DONE

-Report Writing.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:11**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-None

2. WORK TO BE DONE

-Report Writing.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:12**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-None

2. WORK TO BE DONE

-Report Writing.

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLC

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: 3, 3	**Study week no.:13**
Student Name & ID: Wayne Fong 18ACB02229	
Supervisor: Ts Dr Gan Ming Lee	
Project Title: Real-Time Vision-Based Fish Monitoring System Using Raspberry Pi	

1. WORK DONE

-Finsihed and Checked Report.

2. WORK TO BE DONE

-None

3. PROBLEMS ENCOUNTERED

-None

4. SELF EVALUATION OF THE PROGRESS

-None

GMLee

Supervisor's signature

Student's signature

POSTER

REAL-TIME VISION-BASED FISH MONITORING SYSTEM USING RASPBERRY PI

Introduction

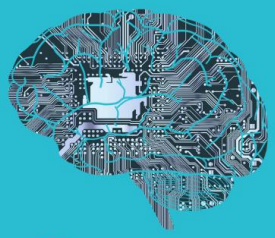
In society, Raspberry Pi has become a very popular computer used in many projects. At the same time, people have been using Raspberry Pi to watch over their fishes in aquaculture farm. Therefore, the final deliverable would be a fish monitoring system powered with Raspberry Pi embedded with deep learning model for fish detection.

OBJECTIVE ?

1. To develop a fish monitoring system that provides visual monitoring services.
2. To allow users to use the system from a distance.
3. To develop a detection module with at least 70% of mAP (mean average precision).
4. To design an electrical circuit for improving the overall monitoring process.

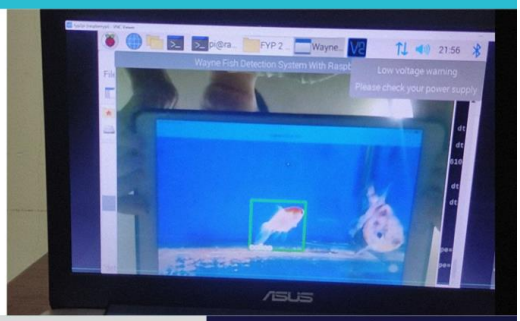
METHODOLOGY

1. DEVELOP DETECTION MODEL.
2. SETUP RASPBERRY PI.
3. EMBED DETECTION MODEL INTO RASPBERRY PI.
4. DESIGN AN ELECTRICAL CIRCUIT FOR IMPROVING THE OVERALL MONITORING PROCESS.



RESULT

If an image is captured and a fish exists in the frame, the system will show the detection output as shown in the right side.



BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) COMMUNICATIONS AND NETWORKING

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

By Wayne Fong **Final Year Project 2**

PLAGIARISM CHECK RESULT

Turnitin Originality Report

Processed on: 18-Apr-2022 02:42 +08
ID: 1812400202
Word Count: 17927
Submitted: 2

Real-time vision-based fish monitoring system... By Wayne Fong

Similarity Index	Similarity by Source	
6%	Internet Sources:	4%
	Publications:	3%
	Student Papers:	2%

include quoted include bibliography exclude small matches mode: quickview (classic) report Change mode print download

<1% match (Internet from 20-Mar-2022)
http://eprints.utar.edu.my

<1% match (Internet from 15-Oct-2021)
http://eprints.utar.edu.my

<1% match (Internet from 07-Jul-2020)
http://eprints.utar.edu.my

<1% match (Internet from 27-Jul-2021)
http://eprints.utar.edu.my

<1% match (Internet from 27-Jul-2021)
http://eprints.utar.edu.my

<1% match (Internet from 27-Jul-2021)
http://eprints.utar.edu.my

<1% match (Internet from 15-Apr-2020)
http://eprints.utar.edu.my

<1% match (Internet from 27-Jul-2021)
http://eprints.utar.edu.my

<1% match (Internet from 24-Jul-2021)
http://eprints.utar.edu.my

<1% match (student papers from 06-Nov-2018)
Submitted to Southampton Solent University on 2018-11-06

<1% match (student papers from 11-May-2017)
Submitted to Southampton Solent University on 2017-05-11

<1% match (publications)
Brian H. Curtin, Suzanne J. Matthews, "Deep Learning for Inexpensive Image Classification of Wildlife on the Raspberry Pi", 2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), 2019

<1% match (publications)
Stewart Watkiss, "Learn Electronics with Raspberry Pi", Springer Science and Business Media LLC, 2020

<1% match (Internet from 21-Dec-2020)
https://www.upgrad.com/blog/basic-cnn-architecture/

<1% match (student papers from 13-Apr-2022)
Submitted to Universiti Tunku Abdul Rahman on 2022-04-13

<1% match (Internet from 10-Nov-2021)
https://fict.utar.edu.my/documents/FYP/FYP2_template/FYP2_Report_Template_CS.docx

<1% match (publications)
Yeong-Hwa Chang, Ping-Lun Chung, Hung-Wei Lin, "Deep learning for object identification in ROS-based mobile robots", 2018 IEEE International Conference on Applied System Invention (ICASI), 2018

<1% match (Internet from 01-Jan-2022)
"Raspberry Pi OS", Wikipedia, en, 2022

<1% match (student papers from 02-Nov-2021)
Submitted to University of Bolton on 2021-11-02

<1% match (Internet from 13-Mar-2022)
https://www.jameco.com/z/KS0001-Keyestudio-Keyestudio-Uno-R3-Development-Board-Arduino-Compatible-_2283863.html

<1% match (Internet from 29-Dec-2017)
http://scholarbank.nus.edu.sg

<1% match (Internet from 15-Nov-2021)
https://sciforum.net/manuscripts/10009/manuscript.pdf

<1% match (publications)
Greg Flurry, "Java on the Raspberry Pi", Springer Science and Business Media LLC, 2021

<1% match (publications)
Eric Gamess, Sergio Hernandez, "Performance Evaluation of Different Raspberry Pi Models for a Broad Spectrum of Interests", International Journal of Advanced Computer Science and Applications, 2022

<1% match (Internet from 16-Jan-2022)
https://zenodo.org/record/5833868/files/SnR_D3.5%20-%20Data-driven%20analytics%20applied%20on%20UAV%20imagery%20using%20deep%20learning_v1.00.pdf?download=1

<1% match (publications)
Eric Gamess, Sergio Hernandez. "Performance Evaluation of Different Raspberry Pi Models for a Broad Spectrum of Interests", International Journal of Advanced Computer Science and Applications, 2022

<1% match (Internet from 16-Jan-2022)
https://zenodo.org/record/5833868/files/SnR_D3.5%20-%20Data-driven%20analytics%20applied%20on%20UAV%20imagery%20using%20deep%20learning_v1.00.pdf?download=1

<1% match (Internet from 12-Jan-2022)
http://sagitario.itmorelia.edu.mx

<1% match (Internet from 14-Sep-2010)
http://wiki.mikkelgade.dk

<1% match (Internet from 04-Feb-2022)
https://www.astesj.com/publications/ASTESJ_030103.pdf

<1% match (student papers from 14-May-2021)
Submitted to Institute of Technology, Nirma University on 2021-05-14

<1% match (publications)
Jeff Cicolani, "Beginning Robotics with Raspberry Pi and Arduino", Springer Science and Business Media LLC, 2021

<1% match (publications)
Ahsan Jalal, Ahmad Salman, Ajmal Mian, Mark Shortis, Faisal Shafait. "Fish detection and species classification in underwater environments using deep learning with temporal information", Ecological Informatics, 2020

<1% match (student papers from 26-May-2016)
Submitted to Yeditepe University on 2016-05-26

<1% match (student papers from 23-Feb-2021)
Submitted to Universiti Putra Malaysia on 2021-02-23

<1% match (publications)
Mariha Afroz, Nazia Hasan, Md. Ishan Arefin Hossain, "IoT Based Two Way Safety Enabled Intelligent Stove with Age Verification Using Machine Learning", 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021

<1% match (student papers from 23-Sep-2018)
Submitted to Universiti Tenaga Nasional on 2018-09-23

<1% match (Internet from 17-Feb-2022)
https://www.aniwaa.com/review/3d-scanners/2021-creaform-handyscan-700-review/

<1% match (publications)
"Neural Information Processing", Springer Science and Business Media LLC, 2017

<1% match (publications)
"Ubiquitous Networking", Springer Science and Business Media LLC, 2017

<1% match (publications)
Aditi Pande, Mousami Munot, R. Sreeemathy, R.V. Bakare, "An Efficient Approach to Fruit Classification and Grading using Deep Convolutional Neural Network", 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), 2019

<1% match (publications)
Mertcan Koçak, Erkin Gezgin, "PARS, low-cost portable rehabilitation system for upper arm", HardwareX, 2022

<1% match (publications)
Nikolay Valov, Irena Valova, "Home Automation System with Raspberry Pi", 2020 7th International Conference on Energy Efficiency and Agricultural Engineering (EE&AE), 2020

<1% match (Internet from 02-Dec-2021)
https://downloads.hindawi.com/journals/cin/2021/5172658.pdf

<1% match (Internet from 15-Dec-2021)
https://staff.najah.edu/media/scientific-research/books/2017/09/30/Facebook-as-an-Educational-Environment-for-Mathematics-Learning.pdf

<1% match (Internet from 26-Oct-2018)
http://www.theseus.fi

<1% match (publications)
Ikram Ben abdel ouahab, Lotfi Elaachak, Fatiha Elouaai, Mohammed Bouhorma, "Chapter 91 A Smart Surveillance Prototype Ensures the Respect of Social Distance During COVID19", Springer Science and Business Media LLC, 2021

<1% match (student papers from 09-Mar-2020)
Submitted to International University - VNUHCM on 2020-03-09

<1% match (publications)
Modupe Odusami, Sanjay Misra, Olusola Abayomi-Alli, Shobayo Olamilekan, Chukwuebuka Moses, "Chapter 5 An Enhanced IoT-Based Array of Sensors for Monitoring Patients' Health", Springer Science and Business Media LLC, 2022

<1% match ()
Trindade, Rui Miguel Marques Pereira, "Implementation of RESTful web-services as a platform for exchanging information between grid operators", 2019

<1% match (Internet from 29-Aug-2017)
https://minerva-access.unimelb.edu.au/bitstream/handle/11343/91707/Millers_Thesis_2016.pdf?sequence=1

<1% match (Internet from 15-Mar-2022)
https://text.123docz.net/document/2746675-measuring-customer-based-centers-evidence-in-hochiminh-city-nguyen-thanh-trung-nguoi-huong-dan-tran-ha-minh-quan.htm

<1% match (publications)
Charles Bell, "Windows 10 for the Internet of Things", Springer Science and Business Media LLC, 2016

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Wayne Fong
ID Number(s)	18ACB02229
Programme / Course	CN
Title of Final Year Project	Real-time vision-based fish monitoring system using Raspberry Pi

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 6 % **Similarity by source** Internet Sources: 4 % Publications: 3 % Student Papers: 2 %	
Number of individual sources listed of more than 3% similarity: 0	

Parameters of originality required and limits approved by UTAR are as Follows:

(i) Overall similarity index is 20% and below, and

(ii) Matching of individual sources listed must be less than 3% each, and

(iii) Matching texts in continuous block must not exceed 8 words

Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

GMLee

Signature of Supervisor

Name: Gan Ming Lee

Date: 21/4/2022

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	18ACB02229
Student Name	Wayne Fong
Supervisor Name	Ts Dr Gan Ming Lee

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
✓	Title Page
✓	Signed Report Status Declaration Form
✓	Signed FYP Thesis Submission Form
✓	Signed form of the Declaration of Originality
✓	Acknowledgement
✓	Abstract
✓	Table of Contents
✓	List of Figures (if applicable)
✓	List of Tables (if applicable)
	List of Symbols (if applicable)
✓	List of Abbreviations (if applicable)
✓	Chapters / Content
✓	Bibliography (or References)
✓	All references in bibliography are cited in the thesis, especially in the chapter of literature review
✓	Appendices (if applicable)
✓	Weekly Log
✓	Poster
✓	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
✓	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)
Date:19/4/2022