DEVELOPMENT OF AUTOMATED PET FEEDING SYSTEM UTILIZING TEACHABLE MACHINE TECHNOLOGY AND RASPBERRY PI

YASMIN NAJIHA BINTI AZMIR

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEVELOPMENT OF AUTOMATED PET FEEDING SYSTEM UTILIZING TEACHABLE MACHINE TECHNOLOGY AND RASPBERRY PI

YASMIN NAJIHA BINTI AZMIR

This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Electronics Engineering Technology (Telecommunications) with Honours

Faculty of Electronics and Computer Technology and Engineering

Universiti Teknikal Malaysia Melaka

2025

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJURUTERAAN ELEKTRONIK DAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : Development of Automated Pet Feeding System Utilizing Teachable Machine Technology and Raspberry Pi

Sesi Pengajian : 2024/2025

Saya YASMIN NAJIHA BINTI AZMIR mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

☐	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan.
☑	**TIDAK TERHAD**	

Disahkan oleh:

______________________ (COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: ....

Ts. FAKHRULLAH BIN IDRIS
Jurutera Pengajar DV44
Jabatan Teknologi Kejuruteraan Elektronik dan Komputer
Fakulti Teknologi dan Kejuruteraan Elektronik & Komputer
Universiti Teknikal Malaysia Melaka

Tarikh : 26/1/2025 Tarikh : 26/1/2025

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled "DEVELOPMENT OF AUTOMATED PET FEEDING SYSTEM UTILIZING TEACHABLE MACHINE TECHNOLOGY AND RASPBERRY PI" is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	
Student Name	:	YASMIN NAJIHA BINTI AZMIR
Date	:	28 JANUARY 2025

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Electronics Engineering Technology (Telecommunications) with Honours.

Signature :

Supervisor Name : TS. FAKHRULLAH BIN IDRIS

Date : 28 JAN 2025

DEDICATION

Alhamdulillah, praise to the Almighty Allah S.W.T

This project is dedicated to:

My mom Zuraini Ibrahim,

My beloved family,

My Supervisor Ts Fakhrullah Bin Idris

And all my friends Izzaty, Auni, Nurin, Natasha, Afiq and Syazwan,

Thanks for supporting me all the way.

ABSTRACT

Cats are seen as clean which is making it safe to keep them. With the increased use of technology and a hectic lifestyle, individuals seek emotional bonds that can offer them delight. Cats are loyal and caring, which is why many people choose to adopt them. Cats are like true members of the family, however there are occasions when owner need to leave their cats at home for an extended period. Cats require special care, which is not as easy as it used to be with today's hectic lifestyle. The development of automatic pet feeders aims to simplify pet owners' responsibilities by allowing them to feed their pets anytime and anywhere. The system's core component is IoT-based and built with Raspberry Pi, allowing it to care for their cats at home on its own. The system is coupled with the Pi camera, which allows to shoot pictures over the IOT platform and obtain on-demand footage at house. The SD card is an essential component of the Raspberry Pi, providing internal storage for data. Controls a servo motor to distribute precise amounts of food, reducing wastage and ensuring that cats are properly fed. This system is particularly beneficial for cats, which often require multiple small meals throughout the day. The automated pet feeding system successfully differentiates between the owner's cat which is Lulu, and other stray cats or objects around the feeder with an accuracy of 99.9%. This precision ensures that food is dispensed only for Lulu, reducing wastage, and preventing misuse. This advanced technology ensures that cats receive the care they need, even when their owners are away, combining convenience and peace of mind in pet care.

Keywords: Pi camera, Raspberry pi, Pet feeder, Teachable Machine

ABSTRAK

Kucing dilihat sebagai bersih yang menjadikannya selamat untuk memeliharanya. Dengan peningkatan penggunaan teknologi dan gaya hidup yang sibuk, individu mencari ikatan emosi yang boleh memberi mereka kegembiraan. Kucing setia dan penyayang, itulah sebabnya ramai orang memilih untuk mengambilnya. Kucing adalah seperti ahli keluarga yang sebenar, namun ada kalanya pemilik perlu meninggalkan kucing mereka di rumah untuk tempoh yang lama. Kucing memerlukan penjagaan khas, yang tidak semudah dahulu dengan gaya hidup yang sibuk hari ini. Akibatnya, salah satu isu utama telah menentukan cara membangunkan kucing dengan kaedah yang mudah. Pembangunan penyuap haiwan kesayangan automatik bertujuan untuk memudahkan tanggungjawab pemilik haiwan peliharaan dengan membenarkan mereka memberi makan haiwan peliharaan mereka pada bila-bila masa dan di mana sahaja. Komponen teras sistem adalah berasaskan IoT dan dibina dengan Raspberry Pi, membolehkannya menjaga kucing mereka di rumah sendiri. Sistem ini digabungkan dengan kamera Pi, yang membolehkan untuk menangkap gambar melalui platform IOT dan mendapatkan rakaman atas permintaan di rumah. Kad SD ialah komponen penting Raspberry Pi, menyediakan storan dalaman untuk data. Mengawal motor servo untuk mengagihkan jumlah makanan yang tepat, mengurangkan pembaziran dan memastikan kucing diberi makan dengan betul. Sistem ini sangat bermanfaat untuk kucing, yang sering memerlukan beberapa hidangan kecil sepanjang hari. Sistem pemberian makanan haiwan peliharaan automatik ini berjaya membezakan antara kucing pemilik yang bernama Lulu, dan kucing liar atau objek lain di sekitar pemakanan dengan ketepatan 99.9%. Ketepatan ini memastikan bahawa makanan hanya diberikan untuk Lulu, mengurangkan pembaziran, dan mencegah penyalahgunaan.Teknologi canggih ini memastikan bahawa kucing menerima penjagaan yang mereka perlukan, walaupun pemilik mereka tiada, menggabungkan kemudahan dan ketenangan fikiran dalam penjagaan haiwan peliharaan.

Kata Kunci: Kamera Pi, Raspberry Pi, Pet feeder

ACKNOWLEDGEMENTS

First and foremost, I would like to take a deepest appreciation to my supervisor Ts Fakhrullah Bin Idris from the Faculty of Technology and Electronic and Computer of University Technical Malaysia Melaka for his for his invaluable supervision, patience, inspiration, passion, and extensive knowledge in completing this project.

I would also like to express my heartfelt gratitude to my classmates for their ongoing assistance throughout the completion of this project.

I am also indebted to my mother, a million of thanks to them because of their support to me with their prayer and their love. Finally, I would like to thank all my friends BERT who have been such wonderful friends to me and to everyone else who was involved in the completion of this project. I would like to thank them for all the support and encouragement to me which have given me the courage and wisdom to fulfil my final year project. Thank you.

TABLE OF CONTENTS

LIST OF TABLES

Type text here

LIST OF FIGURES

LIST OF ABBREVIATIONS

kg	-	Kilogram
%	-	Percentage

CHAPTER 1

INTRODUCTION

1.1 Introduction

This chapter will explain the project with its background, problem statement, objective, project scope, project significance and thesis outline.

1.2 Background

Pets, as defined, are companion animals with an owner who cares for them and keeps them company, such as cats. Nowadays, the use of technology in everyday life has become extremely frequent. Nearly every operation and activity are carried out making use of technology. Taking care of pets may also be done using technology. This demonstrates that technology play a significant role, and it is impossible not to employ them. Internet of Things (IoT) is one of the most popular technologies now in use. IoT can assist owners in efficiently caring for their dogs while avoiding the stress of a hectic schedule. The Internet of Things allows owners to watch their pets at home from anywhere.

Raspberry Pi plays a crucial role in capturing their pets' faces and triggering the servo motor to revolve the spinner, releasing the pet food. Importantly, a server for accepting control signals from mobile applications was constructed and implemented on the Raspberry Pi. The system is operated by the owners using a smartphone application.

According to Canadian research, the most prevalent reasons for not owning a pet were the requirement to care for the pet when travelling (34.6%), the need for time (28.6%), and the need for reasonable housing (28.3%), with a hatred of pets being less common (19.6%). [1]. Based on the percentages, it can be concluded that humans desire to retain pets but are scared they will not be able to give them their entire attention due to other activities and time constraints while away from home. Thus, there are several hazards associated with keeping pets alone, including weight, stress, and unpleasant interactions. At the same time, pet day-care may not be suitable for our pet's behaviour, and these kinds of places are typically completely booked over the holidays.

In Malaysia, several studies have identified common reasons for not owning pets, like findings in other countries. One significant study noted that the lack of pet-friendly accommodation is a major deterrent. Many landlords and housing developments do not allow pets, making it difficult for potential pet owners to find suitable living arrangements [2]. Additionally, the high cost of veterinary care and limited access to pet-friendly public spaces also discourage pet ownership.

The overall function of the system in the form of feeding the pets either directly or scheduled, as well as capture of photographs around the feed. This Automatic Pet Feeder is design to dispense pet food. Servo motor, sensor and Raspberry Pi Camera Module that connected to the Raspberry Pi which will be able to control by a web application.

1.3 Project Relation to Current Issues

An automatic pet feeder project aligns with Sustainable Development Goal 12: Responsible Consumption and Production. One important connection is the promotion of responsible consumption and production, one of the SDGs [3]. An automatic pet feeder can help reduce food waste and ensure that pets receive the proper amount of food at the right time, aligning with the goal of responsible consumption and production. Furthermore, such a project can contribute to the goal of sustainable cities and communities by meeting the requirements of urban pet owners and improving the well-being of pets in urban surroundings, encouraging a more sustainable and pet-friendly community. This project contributes to SDG 12 targets like halving per capita food waste and reducing waste generation, making a positive impact on responsible consumption and production practices.

1.4 Problem Statement

- In pet hotels, there are lots of animals, which can make it hard to give each one special attention. Staff might not always understand what makes the pet unique, like what they like or do not like [4]. So, it can be tough to make sure the pet gets the care they need. It is like finding a special place among many choices.
- For some folks, the thought of pets staying in hotels can be a bit of a bother. They might have allergies that act up around animals, or they simply do not like the idea of sharing a space with furry friends [5].

- Cats are often grazers and may prefer to eat small amounts of food throughout the day rather than large meals. Leaving food out allows them to eat as they please [5].
- Hotels are notorious for the spread of contagious diseases like kennel cough, canine distemper, rabies, parvovirus, and canine influenza. Diseases are spread through direct contact and sharing contaminated items [6].
- Many pet owners have difficulty feeding their pets regularly due to busy lifestyles and increased responsibilities. This can lead to pets being neglected or not fed properly. [6]

1.5 Objective of the Project

I. To develop a system that automates the pet feeding process using Raspberry Pi.

II. To demonstrate the capability of the Raspberry Pi Camera Module to capture images effectively using image recognition to dispense correct amount of feed to specific pet.

III. To design a user-friendly interface for the system, it is essential to analyse user interactions and ensure the platform is intuitive and easy for pet owners to use.

1.6 Scope of Project

This project revolves around the development of an innovative Automated Cat Feeder system, meticulously designed to cater to the needs of small pets such as cats. The area of the scope also focuses on cat at home. This automatic pet feeder only supplies dry food and no water. Considering the specific requirements of cats, the feeder utilizes a compact container for food storage, ensuring that it remains appropriate in size for the intended users. One of the primary objectives of the project is to provide users with the convenience of wireless connectivity, enabling seamless interaction with the Automated Cat Feeder system through various devices.

Raspberry pi work together with Teachable Machine and Infrared Sensor to detect the level of food in the bowl. To avoid the overflow of food in the bowl. This level of customization ensures that each cat receives the appropriate amount of food at the right time. Servo motor will be used to dispense the food into the bowl. After Teachable Machine connect with Raspberry Pi to send data, the Raspberry Pi will be communicated with motor then dispensed the dry cat food. In essence, this project aims to revolutionize the way cats are fed, combining cutting-edge technology with thoughtful design to enhance the lives of both pets and their owners.

Lastly, the system exclusively dispenses dry food, adhering to dietary preferences and nutritional recommendations for cat diets. Recognizing the importance of reliability in pet care technology, the Automated Cat Feeder system is engineered for long-term operation, with robust construction and dependable performance to ensure continuous functionality over extended periods.

1.7 Project Significance

In today's world, time has become an important aspect of daily life, so people must race against the clock since they do not have enough time to do everything. Consequently, people look for a new product or technology that will make their jobs easier and save them time, particularly as pet owners. Therefore, the innovation of Automatic pet feeder using Raspberry Pi, will become crucial because it will be able to provide benefits to both pets and their owners. This is because the pet can be fed on time, while the owner saves time and energy. Finally, this project focuses primarily with the performance of the Raspberry Pi, which depends on programming and the servo motor. The project's aims will be met thanks to the Raspberry Pi's excellent performance and the servo motor's efficiency.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

New technologies, such as an Automatic Pet Feeding based on Raspberry Pi, will be implemented to help pet owners manage their pets. For this chapter, some research was conducted on materials connected to the scope of this project's work and technique to gather ideas and information about it.

2.2 Understanding Current Issues in the Literature

Through a review of the literature on busy pet ownership and pet care technologies, this literature will explore the ways in which automatic pet feeders can contribute to the understanding of the impact of lifestyle on pet care. This literature has shown that automatic pet feeders can provide valuable insights into the daily habits and routines of busy pet owners.

2.3 Cat Care

Cats are a species that has been conquered for many years, making them quite popular as pets today. In the past, cats were utilised to manage pests and rodents [7]. Cats are now often regarded as members of the family. There are various crucial things to consider while caring for a cat.

2.3.1 Important Cat Care

- **Food and Drink**: One of the most crucial things is to provide them with food and water, particularly water. In a personal discussion with one of the veterinarians in Bandung, he stated that a cat can die without water in three days. The cat's owner must ensure that the cat is always fed on a daily basis. A day without meals can be detrimental to cat health [8]. To lessen obesity, the owner must examine the amount and timing of feeding cats based on their needs [9]. A cat's food demands are around 2% of its body weight [9]. Cats also have a habit of eating frequently with modest amounts of food in each.

- **Environmental Needs**: According to a veterinarian from Bandung, cats are creatures that exhibit strong environmental tendencies. It implies cats will be agitated if their surroundings change dramatically. Cats have certain behaviours, and all of them should be satisfied in the correct surroundings. These habits include hunting, hiding when terrified, staying in high places, and sharpening their nails [9].
- **Health**: To keep their cats healthy, cat owners should vaccinate them. Vaccinating our cats helps prevent various illnesses, including panleukopenia, flu, the FeLV virus, and Chlamydophila bacteria [10]. Cats are also strongly advised to be sterilised to avoid undesired behaviour such as fighting with other cats or spraying [10].

2.4 Automatic Feeding Processes

Traditional pet feeding systems often rely on manual intervention, which can be challenging for pet owners with busy schedules. The incorporation of Raspberry Pi into IoT pet feeding systems enables automation of feeding processes. These systems can be programmed to dispense precise portions of food at scheduled intervals, ensuring that pets receive timely and appropriate meals even when their owners are away. This automation not only promotes consistency in feeding schedules but also reduces the reliance on human presence for basic pet care needs.

IoT-enabled pet feeding systems with Raspberry Pi integration have the capacity to adjust feeding quantities based on pet weight, age, and dietary requirements. This personalized approach to feeding enhances the overall health and well-being of pets by ensuring they receive nutrition tailored to their specific needs [11]. Moreover, the ability to program multiple feeding times throughout the day promotes healthy eating habits and prevents overfeeding.

In conclusion, the incorporation of Raspberry Pi into pet feeding systems signifies a transformative shift towards smart and efficient pet care solutions. As technology continues to evolve, these advancements hold great promise for improving the lives of pets and their owners, fostering stronger bonds, and ensuring the well-being of beloved companions.

2.4.1 Self-Replenishing Feeders

Pet food in the bowl is always full because self-replenishing feeders, like the one shown in Figure 2.1, allow the food to be pushed down by gravity when the pet is eating part of it. To operate this system, the pet meal is typically housed in a sizable container above the bowl. Even though this system is limited to dry pet food, its self-replenishing feature makes it useful even when the owner is not there. The main disadvantage of this approach is that pets—especially those who are obese—should not use it since their food intake needs to be regulated.

Figure 2.1 A Sample of Self-replenishing Feeder

2.5 The Dispensing Food

The use of Raspberry Pi in IoT pet feeding systems enables precise and controlled food dispensing. Servo motors or other mechanical mechanisms connected to the Raspberry Pi board can accurately dispense specified portions of food, eliminating the risk of overfeeding or underfeeding. This level of control is particularly beneficial for pets with dietary restrictions or those requiring specific feeding regimens.

The integration of Raspberry Pi allows for customization of feeding schedules and portion sizes through user-friendly interfaces [13]. Pet owners can adjust feeding parameters remotely using smartphone applications or web interfaces connected to the IoT system. This flexibility ensures that feeding routines can be modified as needed, accommodating changes in pet behaviour or dietary requirements.

In conclusion, the use of Raspberry Pi in IoT pet feeding systems empowers pet owners with greater control and flexibility over their pets' feeding regimens, ultimately contributing to the health and well-being of their beloved companions. As technology continues to evolve, these advancements hold great promise for revolutionizing pet care and fostering stronger bonds between pets and their owners.

2.5.1 Dry Food Portion Based on Cat Size

When it comes to feeding our pet, understanding how much food they need is essential for their health and well-being. Cats come in various sizes, from small kittens to large adults, and their portion requirements can vary accordingly. In this table, categorized cats based on size and provided approximate portion sizes to help cat owners gauge how much food to give. These portion sizes serve as general guidelines, but it is important to remember that individual cats may have unique dietary needs based on factors like age, activity level, and overall health [14].

Cat Weight	Daily Portion of Dry Food (cups)
Up to 2 kg	1/4 to 1/3
2 to 3 kg	1/3 to 1/2
3 to 4 kg	1/2 to 2/3
4 to 5 kg	2/3 to 3/4
5 to 6 kg	3/4 to 1
6 to 7 kg	1 to 1 1/4
7 to 8 kg	1 1/4 to 1 1/2
8 to 9 kg	1 1/2 to 1 3/4
9 to 10 kg	1 3/4 to 2

Table 2.1 Table of Cat Sizes and Daily Food Portions

This table provides average guidelines based on a cat's weight, but individual needs may vary by up to 50% in either direction [15][16]. Factors like age, activity level, and health status should also be considered when determining portion sizes [16].

2.6 Technological Advancements in Pet Care

The pet care sector has seen a rise in technology developments, changing how the owner of pet take care of their cats. One of the most known inventions is using the camera in automated pet feeder [18].

2.6.1 Cat Recognition

The primary component is the use of cat facial recognition technology to identify specific cats. This requires pre-registering photographs of each cat's face in the feeder system. A software platform where has facial recognition can then identify between the registered cats based on their distinct facial traits. Once a cat is detected by the facial recognition technology, the feeder can dispense the right portion of food for that cat. This prevents the food from waste [32].

2.7 The Important of Hygiene and Food Quality

Food cat is also similar with our food. Automatic pet feeders help maintain the freshness and quality of pet food by dispensing it when the cat come near the camera. This prevents the food from sitting in the bowl for extended periods, which can lead to spoilage, especially in hot weather [19]. In this project, built-in sensors are used to monitor the food in the bowl. This ensures that pets always have access to high-quality, fresh food, promoting their overall health and well-being.

Bacterial Growth	Dry cat food stored at room temperature creates an excellent environment for germs to proliferate quickly.
Oxidation	Dry cat food that has been exposed to air for an extended period can turn mouldy and rotten because of lipid oxidation.
Moisture Absorption	Dry food left out can collect moisture from the air, causing it to become mushy and stale.
Insect and Rodent Infestation	Open pet food dishes left unattended can attract insects, rats, and other pests.
Loss of Nutrients	Cat food nutrients can degrade over time because of exposure to air, light, and heat.

Table 2.2 Effect of Leaving Cat Food

2.8 Related Research

Citation	Title	Problem	Method
[1]	Development of Cat Care System Based on Internet of Things	Cat owners are not constantly there with their cats; thus, cat care cannot be carried out efficiently. Thus, there are several problems associated with keeping the cat alone, including weight, stress, poor interactions with other cats, and so on. At the same time, cat day-care may not be appropriate for our cat's temperament, and these facilities are typically completely booked throughout the holiday season.	In the context of the internet of things, the components that comprise the complete system are thought to be hardware, software, and communication. The system is separated into numerous sub-systems, including the cat feeder, cat playmate, automated cat door, cat monitor, and mobile applications.
[20]	Design of Mini Pets Feeding Intelligent Home System Based on IoT	The professional pet care and feeding equipment available on the market cannot satisfy the needs of real-time monitoring and remote control, nor can the pet owner take care of the animal from a distance.	The system was built on three layers of Internet of Things architecture: sensor, network, and application. CC2530 served as the bottom computer's coordinator, completing data gathering of the LAN network using the ZigBee wireless transfer protocol. The STM32 microcontroller served as the lower computer's core controller, controlling the interface circuit for each execution component. The ESP8266 Wi-Fi communication module was installed on the STM32 to create an embedded gateway, completing data transmission between the bottom computer, higher computer, and mobile intelligent terminal.

[21]	Design and Implementation of a Pet Care System	Not stated	A setup of Raspberry Pi, a servo, and a food dispenser were linked to create a pet feeding machine. The pet's food might be released from the food dispenser as a result of Raspberry Pi activating the servo and rotating the spinner. ReactJS was used to create a progressive web application that allows you to remotely manipulate the pet feeding equipment. Importantly, a web server for accepting control signals from the web app was written and installed on the Raspberry Pi using NodeJS. This system included core aspects including instantaneous feed and scheduled feed.
[18]	Mobile based monitoring system for an automatic cat feeder using Raspberry Pi.	Having a cat at home requires time and effort.	The webcam can capture images (photographs or movies) and process them using the Raspberry Pi's fswebcam and avconv routines. The stepper motor can spin the feed valve using a Raspberry Pi's General Purpose Input Output (GPIO) pin and a programme. The Raspberry Pi will then be linked to the Internet and a server network, allowing for remote system control via a web browser or mobile web view. The system's overall job is to feed the cat, either directly or on a schedule, as well as monitor images or videos around the feed.

Table 2.3 Comparison of the Existing Research

Table 2.3 presents an ongoing comparison of recent initiatives that focus on creative ideas for automatic pet feeders. Each initiative brings a new perspective and innovations in technology to the table, representing the field's progress in recent years. In the following subtopics, we will look at the highlights and breakthroughs of these projects, revealing how they have jointly pushed the boundaries of pet care technology and established new standards for future advances.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter provides details on the approach and perspectives that were employed to investigate and determine how closely the project aligns with current theories and research. The hardware and software components of this project are being implemented. Testing will be done to identify the project's problem if any errors arise that prevent the project from achieving its goal. Every study methodology-related hypothesis is expressed in detail. The steps that have been performed to finish this project are known as the project methodology. It will symbolise the entire project's flow. The approach can also serve as a manual for individuals who wish to enhance the project in the future by applying it in additional situations. The software and hardware development are the two components which make out the technique.

3.2 Project Flow

Overall, the primary components of this project are the Raspberry Pi, Software platform and the Smartphone. First, conduct a design thinking session to gain an improved understanding of the project's issue statement through research on users who care for pets. Then, it conducts a literature review, journal and previous thesis project on the automated pet feeder, open sources a Raspberry Pi single board and an Android app on a smartphone for automated pet feeder purposes. The pet feeder system with Android app is then created and programmed using the Raspberry Pi. The design's hardware is then built-in accordance with the project proposal. The project will be incorporated, then tested and validated. The project will be integrated and undergo test and validation. After that, decision will be made either be modified or completed with no error. The design flow of this research is demonstrated in Figure 3.1

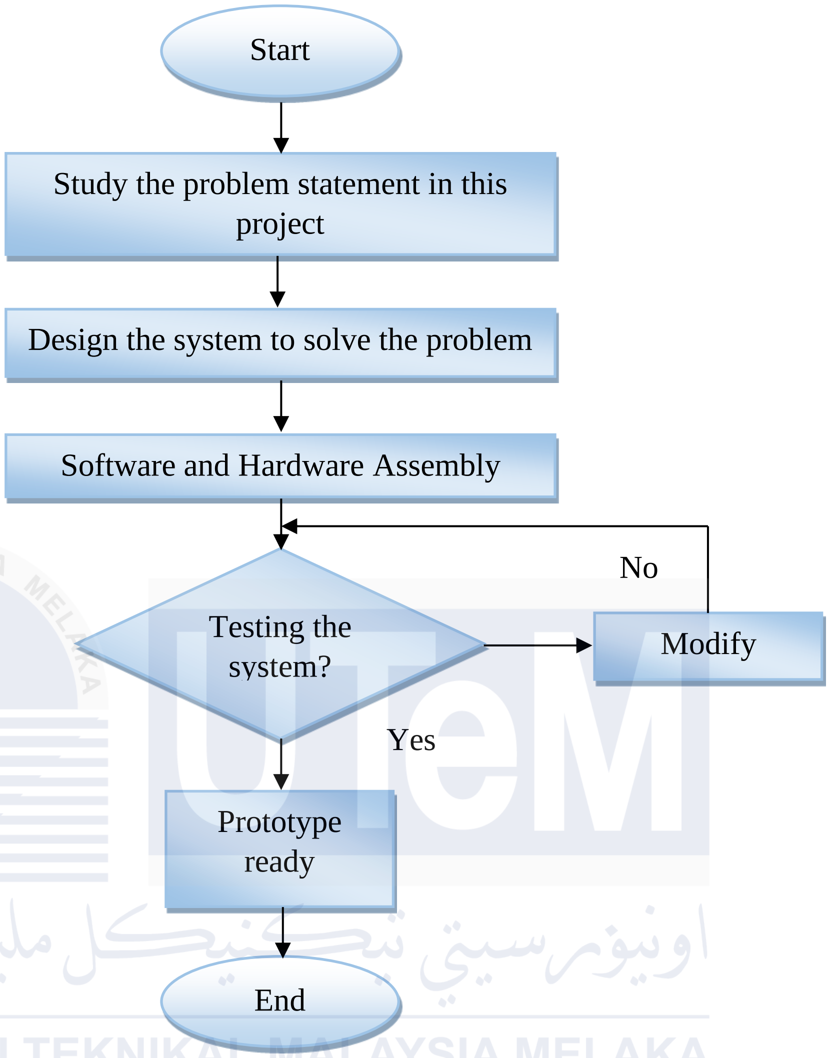


Figure 3.1 Overall Project Flow

3.3 Block Diagram

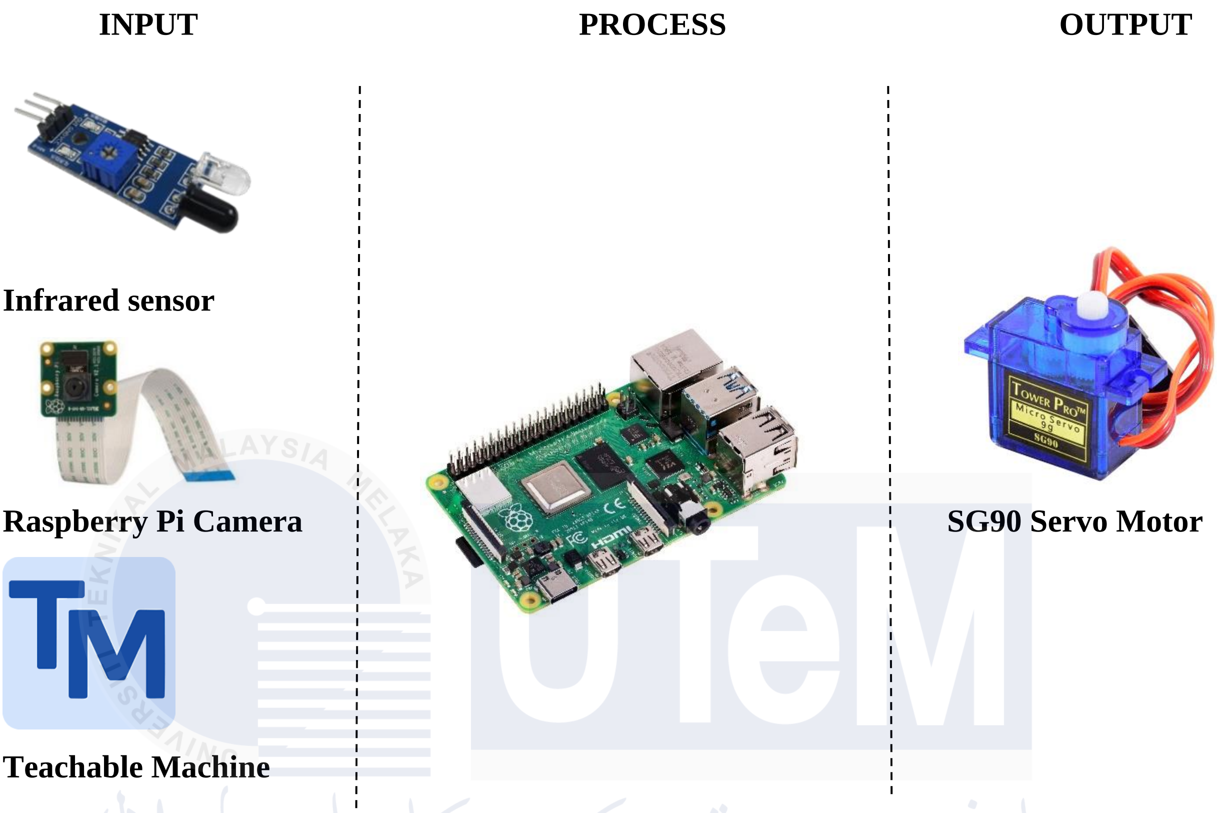


Figure 3.2 Block Diagram of the Automatic Pet Feeder System

Figure 3.2 illustrates the block diagram for the pet feeding system. Aside from flowcharts, block diagrams can be utilised to provide an overview of the project flow. It refers to input on the left, the main processor in the centre, and output on the right. The intricacies of each block diagram will be discussed below.

3.4 Automatic Pet Feeder

As the project progresses on, the most important aspect is organising methodical phases to clearly outline the project's process. The process is shown in an easy-to-understand flowchart, which helps to clarify and simplify the approaches used to complete this project. The flowchart containing all the process planning for this project is projected in Figure 3.1.

Refer to flowchart in Figure 3.2, the system of this pet feeder starts when Teachable Machine detect motion or pet’s face near the pet feeder. If the cats face is detecting in a range of the Raspberry Pi camera, the program will continue to check for infrared sensor in a bowl.

If infrared sensor detects existence of food in the bowl, motor will not rotate. However, if the Teachable Machine detect a movement in a range that have been set and no food detected, motor will rotate. After motor rotate, it will send notification to pet owner by Pushover application said, "Cat Eat”.

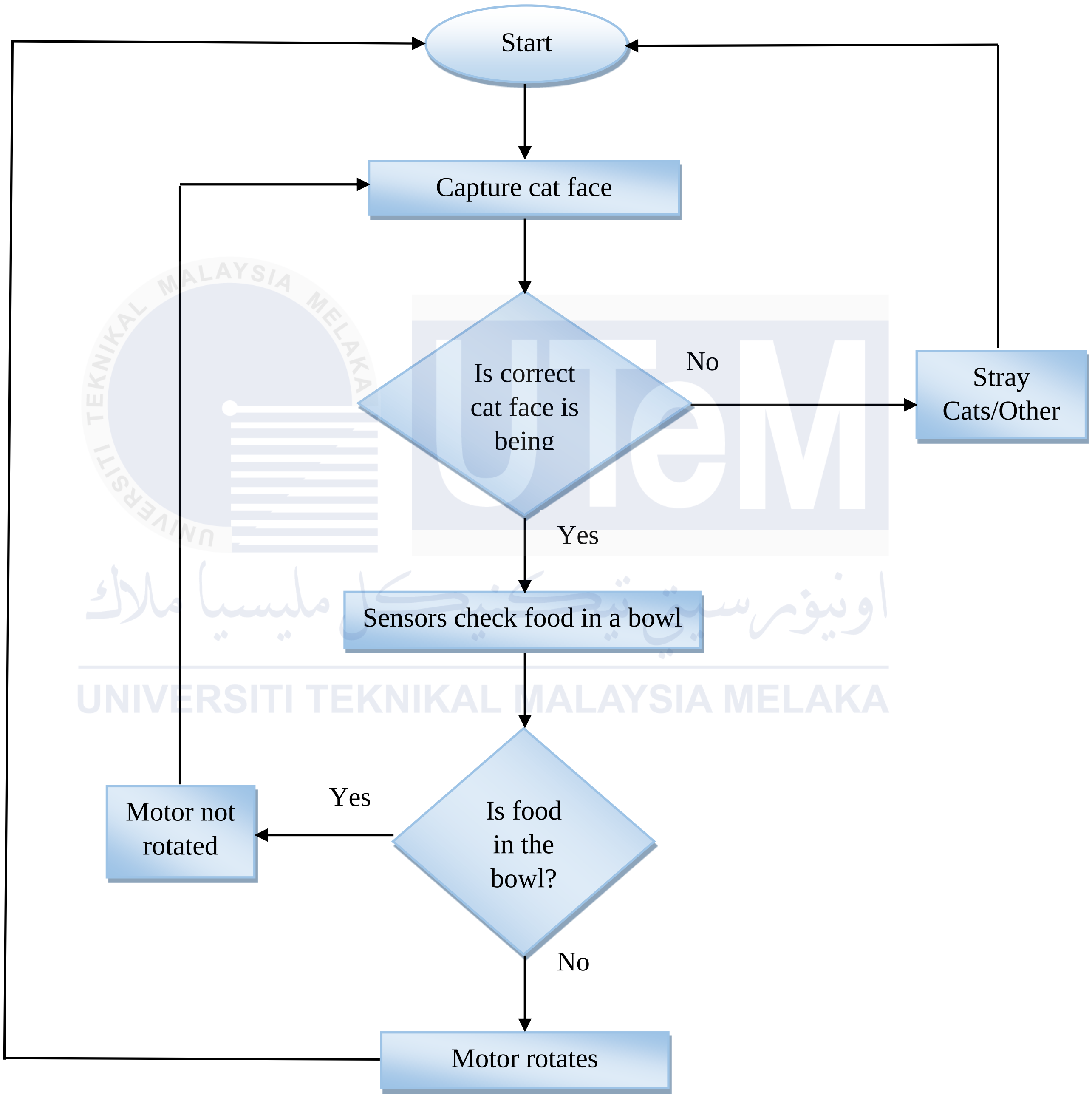


Figure 3.3 Flowchart of Automatic Pet Feeder

3.5 Calculation

For this project, the dry cat food is measured using normal dry measuring cups, which are frequently used for baking. Be sure to be aware that the measurements are based on the food's volume in the cup, not its grammes of weight. Owner can quickly and precisely give the cat the appropriate amount of food by utilising these cups.

Measurement	Grams
1 cup	90
3/4 cup	67.5
2/3 cup	50
1/2 cup	45
1/3 cup	30
1/4 cup	22.5
1 tablespoon	5.6

Table 3.1 Conversion Table for Dry Cat Food

In this project we are using cat weighing 4 kilograms. Then examine the dry cat food's packaging before using it. Typically, it offers a chart, or a suggested feeding quantity determined by the weight of the cat. A typical recommendation is between 40 and 60 grammes where half cup until ¾ cup of dry food per kilogramme of body weight each day, however this can vary depending on the brand [21].

The calculation shows dry food cat eat per day:

$$Total\ daily\ ammount = 4kg\ x\ 50\ grams$$

$$Total\ daily\ amount = 200\ grams$$

In this project, baker measuring cups are utilized,

$$200\ grams = 2\ and\ \frac{1}{4}\ cups$$

3.6 Raspberry Pi

People of all ages may learn computing with the Raspberry Pi, a credit card-sized computer that supports programming languages. It can do nearly everything, including word processing, playing games, watching high quality video, and surfing the internet. However, as software continues to be produced globally and applications become more accessible, there appears to be a decline in computer knowledge. The creator of Raspberry Pi considered rekindling public interest in programming as a solution to this issue. Since the Raspberry Pi was created, people have used it for a wide range of purposes due to its portability, connectivity, programmability, and small size.

3.6.1 Raspberry Pi 4, Model B

The Raspberry Pi 4 is the fourth iteration of the mainline Raspberry Pi single-board computers. Developed by Raspberry Pi Trading and released on June 24, 2019. The single board contains three USB ports, GPIO pin, Ethernet port, Audio/Video port, slot for camera and HDMI port.

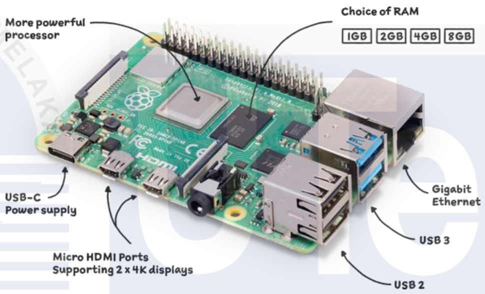


Figure 3.4 Overview of Raspberry Pi 4

Furthermore, Raspberry Pi 4 Model B used ARM Cortex-A72 processor. Pi 4 B models appeared with the improved Broadcom BCM2711C0.

3.6.2 Raspberry Pi Camera

The camera's primarily role is to record photographs of the cat's face. These photos are then processed by a machine learning model (trained with Teachable Machine) to identify and confirm that the detected animal is the right cat.

3.7 Teachable Machine

This project aims to develop an automatic pet feeder that can capture of a cat's face. Teachable Machine are being used, a machine learning platform that enables to build and train custom image classification models. The Teachable Machine model provides the intelligence to recognise when the cat is present and trigger the food dispensing.

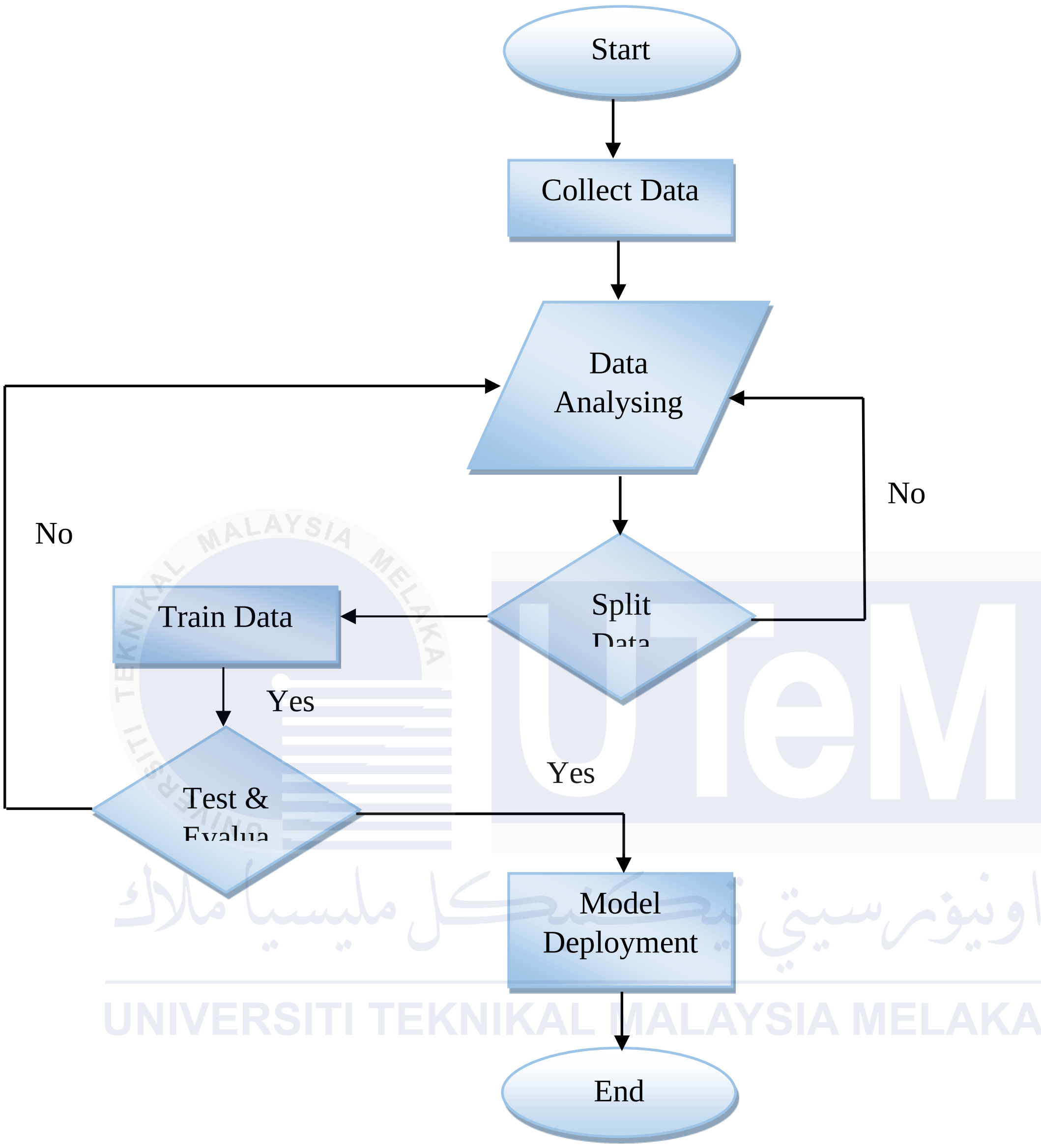


Figure 3.5 Flowchart of Software Platform

3.8 Infrared Sensor

IR Sensors use a specialised light sensor to detect a specific light wavelength in the Infra-Red (IR) spectrum. In this system, an infrared sensor is used as input to detect the presence of food in the bowl of a pet feeder.

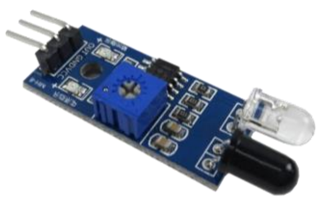


Figure 3.6 Infrared Sensor

3.9 Servo Motor

A servo motor worked to rotate the valve. The valve will allow the food into the bowl. The motor was powered by an external power supply due to the Raspberry Pi's limited power and to prevent damage to the board.

Figure 3.7 Servo Motor

3.10 Teachable Machine

The initial phase of integrating the Teachable Machine model into the automatic pet feeder project focused on evaluating its ability to recognize pets and trigger feeding actions. The model was trained on relevant datasets, tested under various conditions, and assessed for accuracy, reliability, and response time.

3.10.1 Collecting Data Images

1. **Owner's Cat**: Images of the cat that belongs to the owner.
2. **Stray Cats**: Images of cats that do not belong to the owner.
3. **Other Surrounding the Pet Feeder**: Images of objects near the pet feeder.

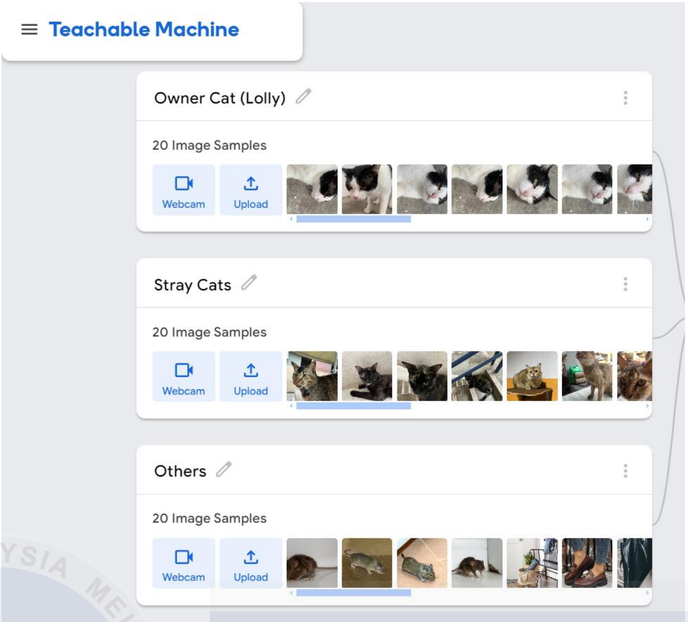


Figure 3.8 Images of owner' cat and the others

From figure 3.8, each class 20 pictures variety of photographs of the cat's face, stray cats and other things has been collected. These photographs were taken from various perspectives and under varying lighting conditions to ensure that the model could reliably identify the cat in a variety of situations.

3.10.2 Training the Model and Evaluating Model Accuracy

After uploading and labelling the photographs, we began the training process. Teachable Machine used these labelled photos to train its neural network model. During training, the model learned to recognise distinctive features and patterns on the cat's face. After training, evaluated the model's accuracy with an additional validation set of photos. This phase confirmed that the model could accurately identify the cat's face. If the accuracy were insufficient, then modify the training data and retrain the model.

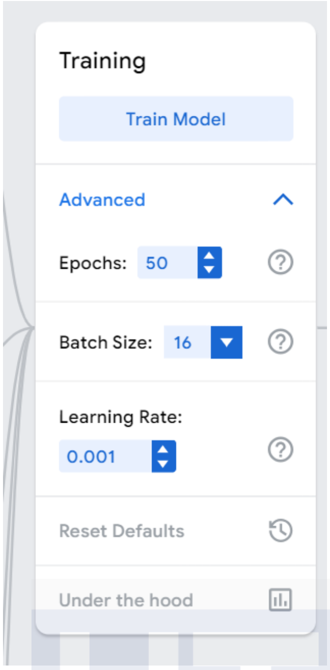


Figure 3.9 Adjustment of Training the Model

3.10.3 The Preview after Model Trained

This section provides a detailed examination of the model's predictions on various sample images. A preview of the trained model was performed to demonstrate its classification capabilities. Each image was fed into the trained model, and the predicted class along with the confidence level (probability) was recorded. For example, this is a preview of 'others' class.

- **Description**: An image of shoes near the cat feeder
- **Model Prediction**: Others
- **Confidence Level**: 100%
- **Analysis**: The model correctly identifies the shoes as not being the owner's cat or a stray cat, with a high confidence level. This demonstrates the model's ability to distinguish between different animals in the vicinity.

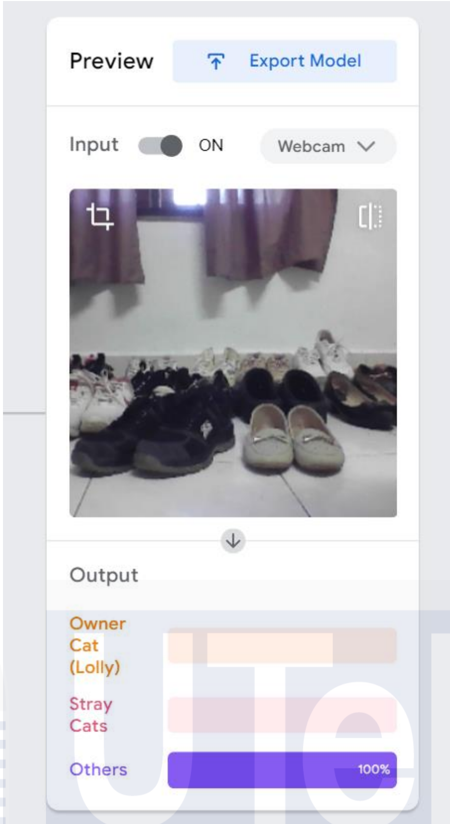
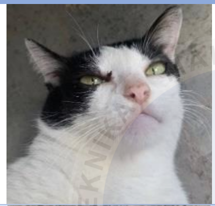
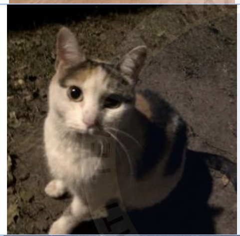
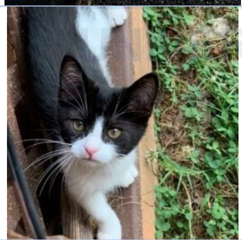
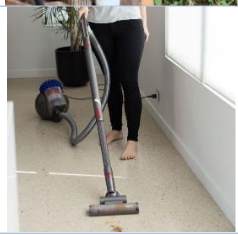


Figure 3.10 Preview of Model

3.11 Preliminary Results

Before deploying a teachable machine learning model with Raspberry Pi, it is essential to evaluate its performance on test data that it has not seen during training. For the model preview, a curated set of 10 test images was selected. These images were carefully chosen to represent a variety of scenarios and potential challenges that the model might encounter in practical use. The images cover different classes, including the owner's cat, other animals surrounding the pet feeder, and potentially misleading objects such as shoes. Each test image was processed through the trained model, and the predicted class along with the confidence level was recorded for analyse.

Test	Random Image	Percentage similarity of Owner's Cat (%)	Percentage similarity of Stray Cats (%)	Percentage similarity of Others (%)
1		99	1	0
2		100	0	0
3		0	98	2
4		0	0	100

5		0	0	100
6		2	79	19
7		0	99	1
8		28	70	2
9		0	0	100

10		0	2	98

Table 3.2 Result of using variable of Cat Face

Table 3.2 shows how Teachable Machine was used to develop a model that classifies random photos into three categories: owner cats, stray cats, and others. The results showed that certain photos could not be categorised with 100% confidence due to similarities between classes. For example, several owner cats were partially identified as stray cats, most likely due to comparable physical characteristics or histories. Similarly, some stray cats were misidentified as owner cats, while other items fell under the "others" group yet occasionally had characteristics with cats, resulting in misclassification. This finding demonstrates the model's limits in recognising slight variations in appearance and context across the categories.

3.11.1 Capturing Image using Camera Pi

This section details the successful installation of an image capturing system that integrates a Raspberry Pi 4 Model B with a Raspberry Pi Camera Module v2. The process encompasses the hardware setup, software configuration, and a step-by-step methodology for image acquisition. Figure 3.13 shows that prototype has been developed that is specifically designed to facilitate the attachment of a Raspberry Pi and a Raspberry Pi Camera Module. The design considers the physical and electronic requirements of both the Raspberry Pi and the camera module, providing a stable and robust platform for image acquisition tasks.

Figure 3.11 Prototype of Raspberry Pi with Camera

Figure 3.14 illustrates the technique, which involves activating the Raspberry Pi's camera interface, utilizing command-line tools for direct image capture, and automating the process with specific commands. The image capture was executed using the command `libcamera-still -t 1000 -o /home/pi/Pictures/image.jpg`. In this command, the `-t 1000` option sets a timeout of 1000 milliseconds, ensuring that the picture is taken almost immediately. The `-o` parameter specifies the output directory, resulting in the image being saved to `/home/pi/Pictures/image.jpg.`

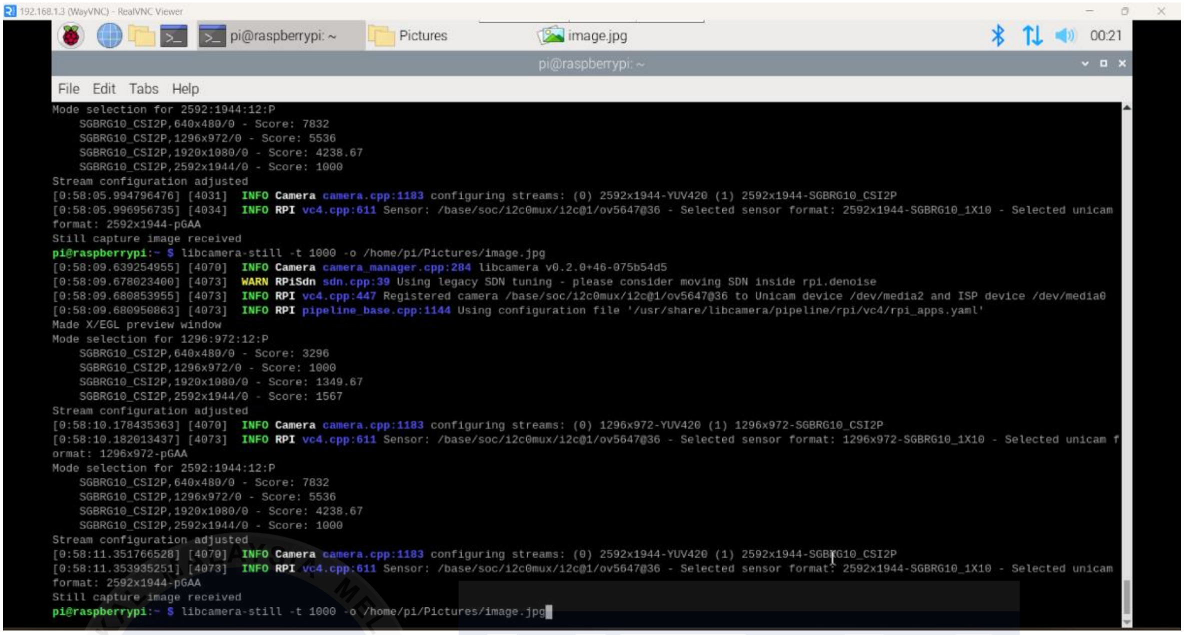


Figure 3.12 Command of Raspberry Pi

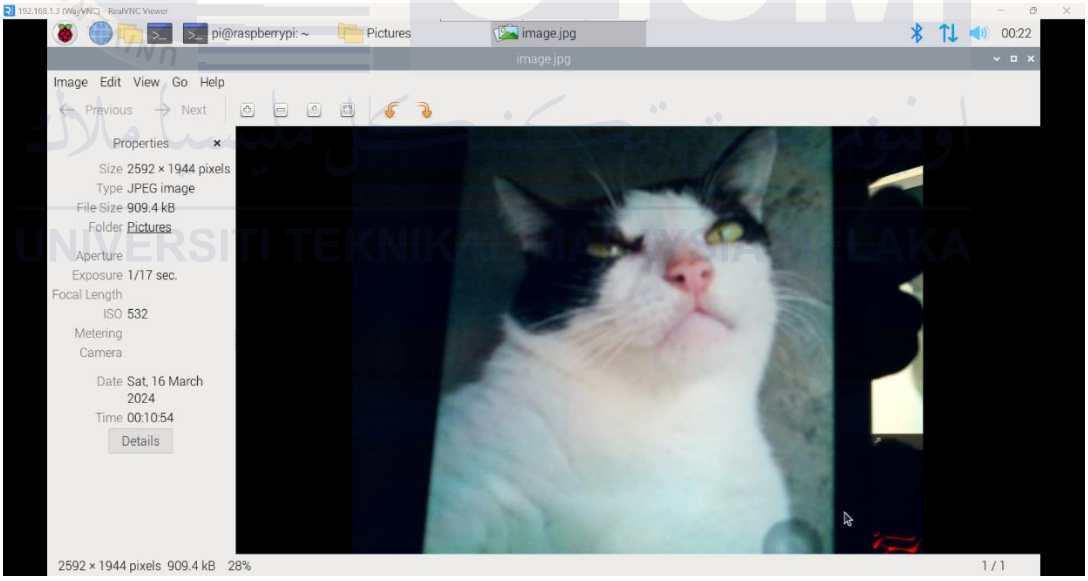


Figure 3.13 Output of Raspberry Pi Camera

The successful execution of this command confirms the proper functioning of the system, as the image was accurately captured and stored in the designated directory. This accomplishment is illustrated in Figure 3.15, which displays the output of the image capture process.

3.12 Conclusion

In conclusion, this chapter outlines the methodology employed in the development of an automatic pet feeder system. It details the project flow, block diagram, and key components including the Raspberry Pi, software platform, smartphone application, infrared sensor, Teachable Machine, and servo motor. The calculation for dry cat food per day is provided, along with insights into the Raspberry Pi 4 Model B, Raspberry Pi Camera, Teachable Machine, infrared sensor, smartphone application, and servo motor. Preliminary results obtained from the Teachable Machine demonstrate its effectiveness in accurately classifying objects related to the feeding process. The system was trained to recognize different gestures or visual cues, such as hand movements or specific patterns, to initiate feeding. Testing revealed that the Teachable Machine achieved a high accuracy rate in detecting pre-defined triggers, ensuring reliable performance in controlling the pet feeder system.

CHAPTER 4

RESULT & DISCUSSION

4.1 Introduction

This chapter will explain on the analysis and discussion about the results obtained throughout the automated pet feeder project whether during and after implementation. It includes the problem faced while continuing the project and the methods to resolve the problem. At the end, this chapter will show the whole result during this final year project. The result of the testing is achieving by the expecting results that have been predicted in the earlier stage of development of the system. The result is comparing several parameters between the simulation and the hardware.

4.2 Prototype

Mechanical design was prioritized and completed first, as it laid the foundation for the hardware development. The design had three primary components: the food container, the food funnel, and the feeder door. Figures 4.1 and figure 4.2 illustrate two design sketches for the project, where Figure 4.1 represents the finalized mechanism, and Figure 4.2 shows the earlier concept, facilitating a comparison of their features. The goal behind this design was collapsibility to maximise space economy when less food was required. Instead, a cylinder container was chosen for its durability and accessibility. As seen in Figure 4.1, the cylinder bottle is used upside down, with the bottom cut off for food access.

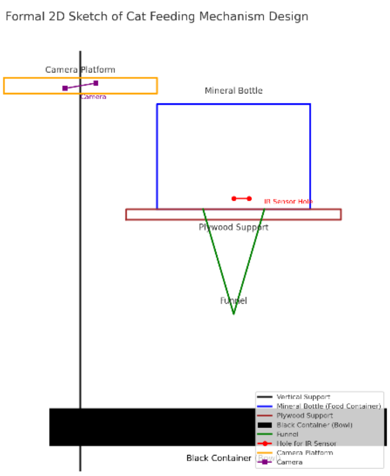


Figure 4.1 Mechanism Design 1

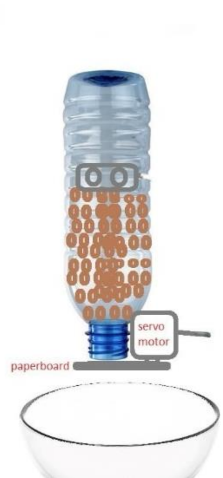


Figure 4.2 Mechanism Design 2

The final mechanical design was relatively simple, consisting of a container, funnel, and a feeder door, which blocks the funnel opening and is actuated by a servo motor in figure 4.3. The materials used in the construction of this prototype consisted of cardboard, a servo motor, a sensor, plywood, and the cylinder water bottle.

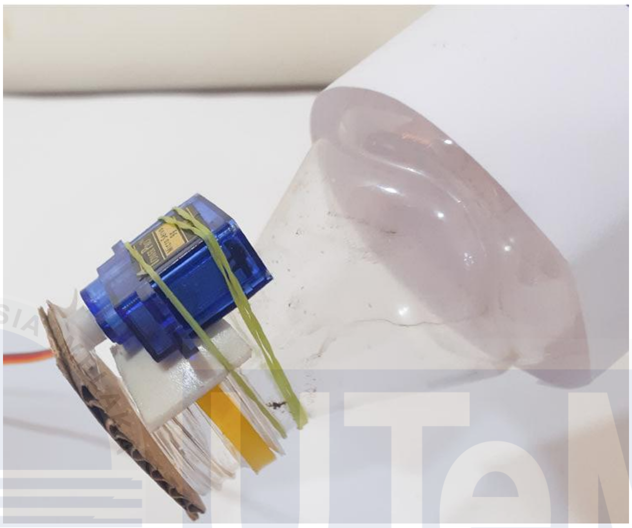


Figure 4.3 Feeder Door

4.3 Project Software

For this project, script Phyton was used infrared sensor, servo motor and camera pi to be function. Both sensor and servo motor will connect to gpio pin in Raspberry Pi.

4.3.1 Phyton Files

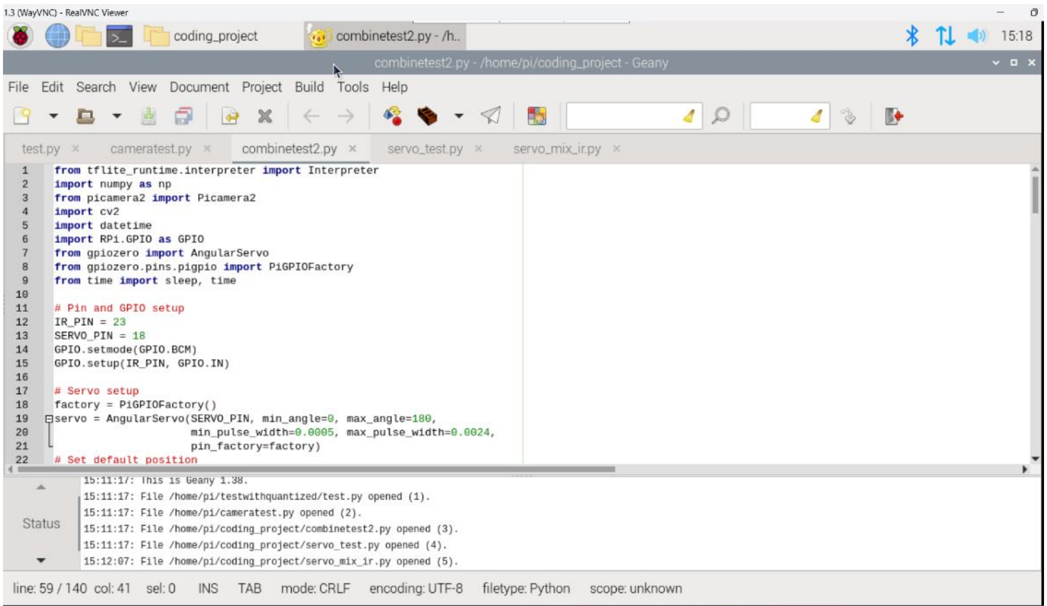


Figure 4.4 Script in Raspberry Pi

Figure 4.4 shown that in Phyton files, there are infrared sensor, servo motor and camera pi files. For infrared sensors, the script combined with servo motor and there is also script for camera pi. Then all the component script will combine altogether. In camera pi, when camera pi detects the owner cat, then servo will not rotate if food still exists in a bowl. This is because servo will only rotate if there is no food in the bowl. If camera pi detects stray cat or others system will automatically will not operate.

4.4 Project Hardware and Electronic Component

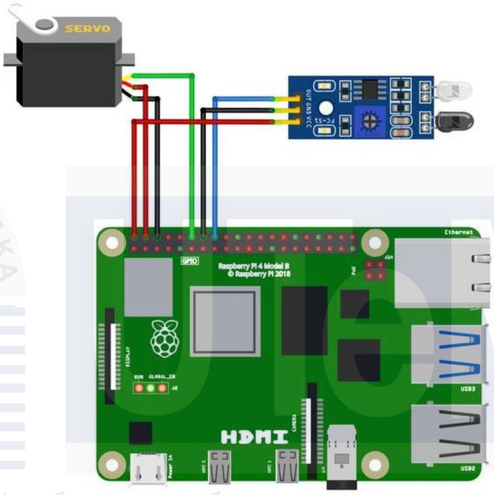


Figure 4.5 Automated Pet Food Dispenser Circuit

Figure 4.5 shown illustrates a circuit that connects a servo motor and an infrared sensor to a Raspberry Pi for an automated food dispensing system. The servo motor dispenses food into a bowl when activated by the Raspberry Pi. The infrared sensor monitors the food level in the bowl, and when it detects that the food is low, it sends a signal to the Raspberry Pi. The Raspberry Pi then triggers the servo motor to dispense more food, ensuring the bowl is always filled.

This project is built using a combination of hardware and electronic components that work together to achieve its functionality. At its core, it features a Raspberry Pi, which serves as the main processing unit, handling computation, data processing, and communication between components. An infrared sensor module is incorporated to enable detection capabilities, such as recognizing cat food. A servo motor is included for precise motion control, allowing the project to perform tasks such as rotating parts of the system with high accuracy. The structural framework of the project is constructed using plywood, chosen for its durability, ease of customization, and ability to support the hardware components securely. Additionally, a bowl is integrated into the design, serving a functional purpose, such as dispensing cat food. From each subtopic below will be discuss the results and difficulties occur during do this project.

4.4.1 Connecting Infrared Sensor with Raspberry Pi

The infrared sensor module, equipped with three pins Output, Ground, and VCC where offers a straightforward and efficient connection to the Raspberry Pi. Its simplicity lies in the direct wiring of these pins to the Raspberry Pi, enabling quick and reliable integration into the system. The sensor is designed to detect the presence of food in the bowl, a crucial functionality in this project. When food is detected, the sensor's onboard LED emits a bright green light, providing an immediate and clear visual indicator.

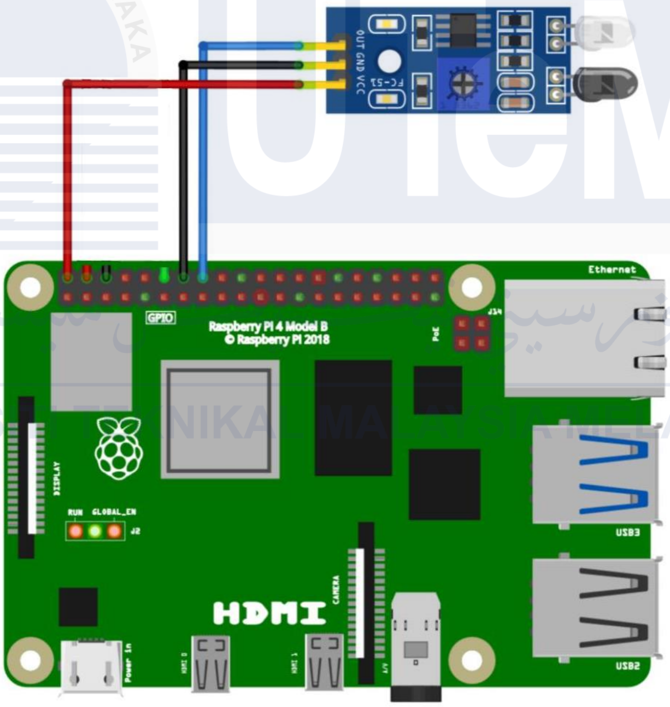


Figure 4.6 Interfacing Infrared Sensor Module circuit diagram with Raspberry Pi

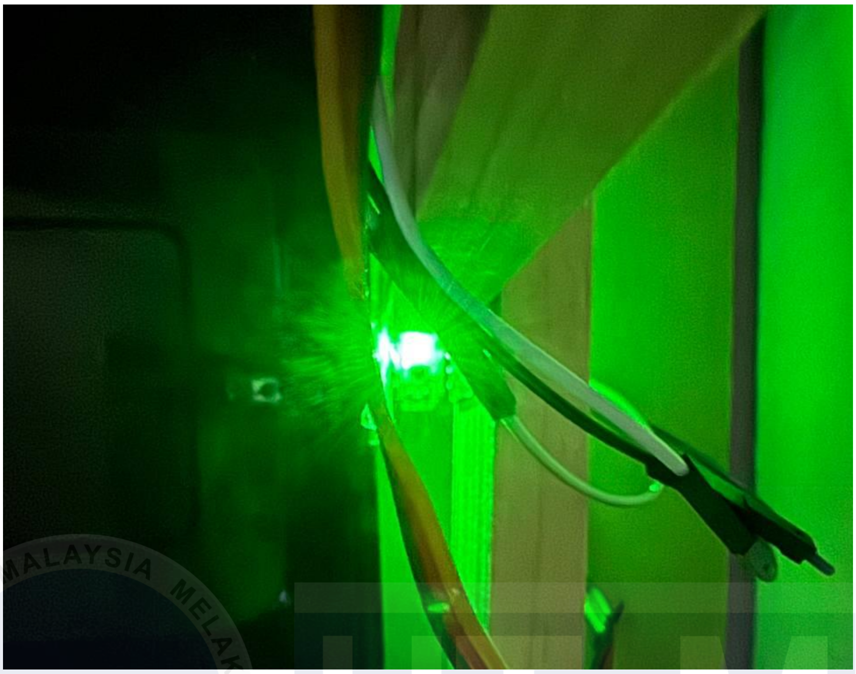


Figure 4.7 One LED of Infrared Sensor light up when no detection of cat’s food

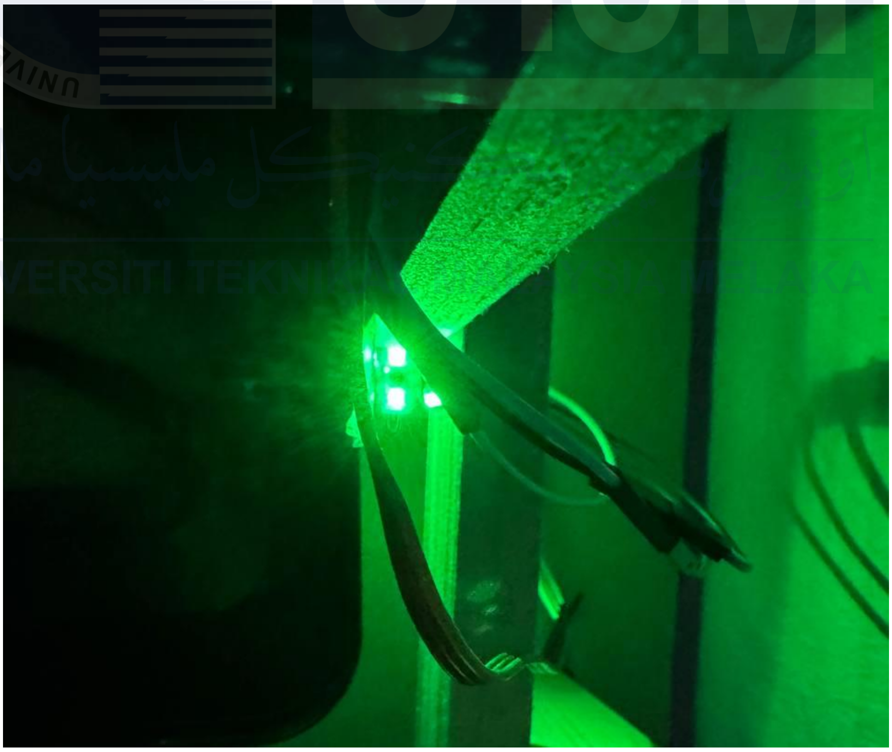


Figure 4.8 Two LEDs of Infrared Sensor light up when there is detection of cat’s food

Figure 4.7 show that no detection because when the sensor does not detect cat food, it lights up one LED. This could indicate a "no food detected" status. It serves as a visual cue that the system is active but not detecting the desired object which is cat food. Figure 4.8 shows that two LEDs means there is detection of cat food), when the sensor detects cat food, it lights up two LEDs. This could indicate a "food detected" status, providing a stronger visual signal that the desired object has been found. The use of two LEDs can signify a more positive or alert state, indicating that the system has successfully identified the presence of cat food.

4.4.2 Connecting Servo Motor with Raspberry Pi

The servo motor, which is seamlessly integrated with the Raspberry Pi, plays a crucial role in the automated feeding mechanism. With its PWM connected to pin 18 in the Raspberry Pi for precise control, the servo motor is programmed to execute smooth and accurate 90-degree rotations. This rotation is carefully calibrated for feeder door to dispense just the right amount of food from the bottle tube into the bowl, ensuring an efficient and consistent feeding process. By leveraging the simplicity of the servo motor's connection and its ability to perform controlled movements, this design achieves a balance between functionality and mechanical reliability, making the motor an indispensable part of the project.

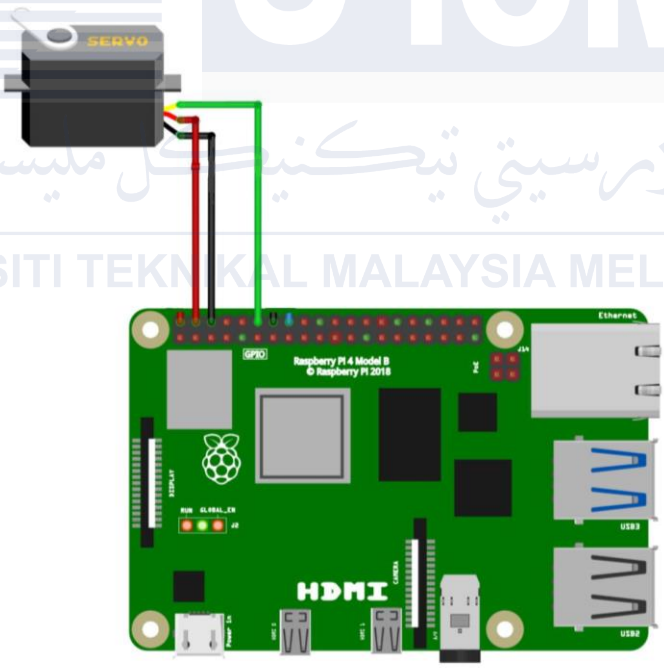


Figure 4.9 Interfacing Servo Motor circuit diagram with Raspberry Pi

Figure 4.10 Servo Motor connect with Feeder Door

Figure 4.10 shows a servo motor connected to a food dispenser system. The servo motor is used to control the food feeder door, allowing it to open and dispense food when activated. The servo motor is attached to a plywood structure, which provides stability and support for the motor. The motor relates to wires to the Raspberry Pi, allowing it to be triggered based on sensor. The plastic bottle above the motor likely holds the food, and the servo controls the mechanism that releases the food when needed.

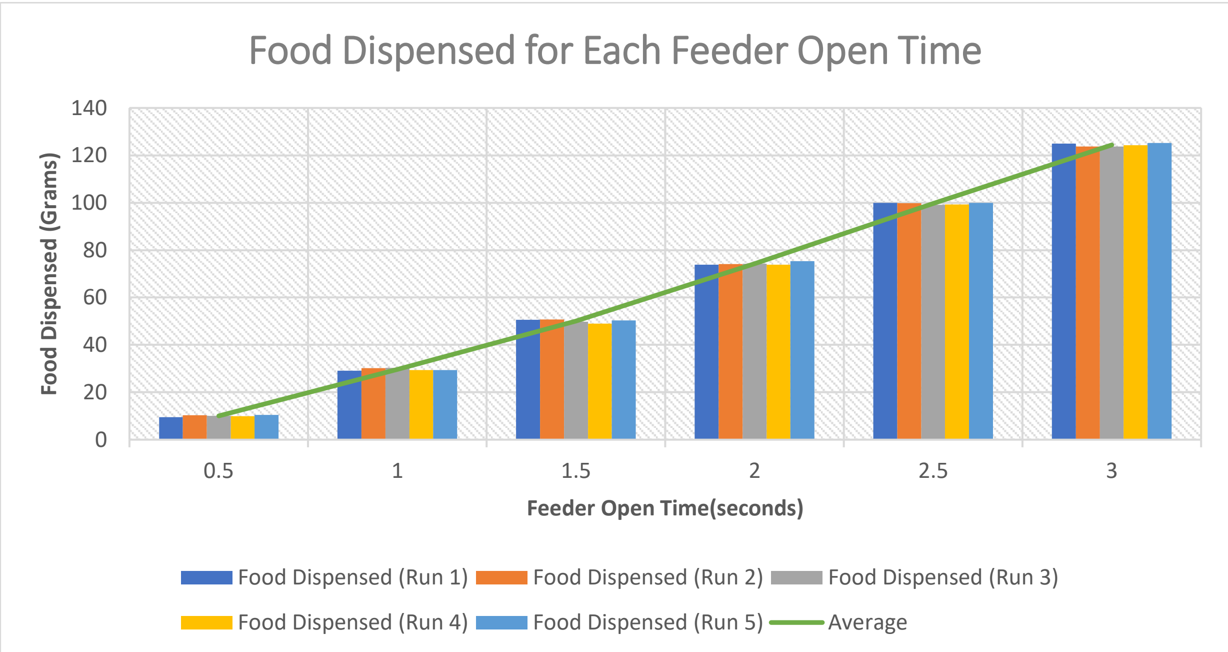
4.4.2.1 Analysis of Servo Operation and Cat Food Dispensing Dynamics

Automated pet feeders have become an essential tool for pet owners, providing convenience and consistency in feeding schedules. The efficiency and reliability of these devices depend on the precise operation of key components, such as servo motors and dispensing mechanisms. This study focuses on analysing the dynamics of servo operation and the flow of cat food in a typical automated feeder system. The aims to bridge the gap between mechanical performance and functional output by examining the interplay between the servo motor's operation and the dispensing system's dynamics.

Feeder Open Time (seconds)	Food Dispensed (Run 1)	Food Dispensed (Run 2)	Food Dispensed (Run 3)	Food Dispensed (Run 4)	Food Dispensed (Run 5)	Average
0.5	9.5	10.3	10	9.9	10.5	10.04
1	29.1	30.2	30.3	29.4	29.4	29.68
1.5	50.6	50.8	49.8	49	50.3	50.1
2	73.9	74.1	74.3	73.8	75.3	74.28
2.5	100	99.8	99.1	99.2	100	99.62
3	125	123.7	123.7	124.3	125.2	124.38

Table 4.1 Feeder Open Time vs. Food Dispensed

The table 4.1 shows the relationship between the Feeder Open Time (in seconds) and the Food Dispensed (in grams) across multiple trials. As the feeder open time increases, the amount of food dispensed also increases. For example, at 0.5 seconds of open time, the food dispensed ranges from 9.5 grams to 10.5 grams, while at 3 seconds, the food dispensed ranges from 123.7 grams to 125.2 grams. The measurements for each open time are relatively consistent, with only slight variations between the trials. This data helps in understanding the feeder's dispensing behavior and can be used to calibrate it for accurate food distribution based on the desired open time.



Graph 1 Food Dispensed for Each Feeder Open Time

In this project, the owner cat weighs 4 kg and requires 200 grams of food daily to maintain a healthy diet. To meet this requirement, the feeder must configure to dispense food in cycles lasting **one second** each, with each cycle dispensing **30 grams of food**. The one-second duration was carefully chosen to ensure smooth and consistent food flow without clogging or spilling. The focus on the one-second timing not only promotes smooth operation of the servo motor but also helps maintain consistency in food dispensing, supporting a balanced feeding schedule.

4.4.3 Connecting Camera Raspberry Pi

The Raspberry Pi camera brought the project to life by successfully detecting the presence of a cat in real-time, thanks to a custom-trained model developed with Teachable Machine. Using machine learning, the model was taught to recognize the cat's features with remarkable precision, enabling seamless integration into the automated feeder system. Whether the cat was lounging, moving, or in varying lighting conditions, the camera consistently identified it, ensuring accurate monitoring and interaction. This exciting combination of Teachable Machine's user-friendly training platform and the Raspberry Pi's versatility opens endless possibilities for smart pet care and beyond.

Figure 4.11 Raspberry Pi Camera

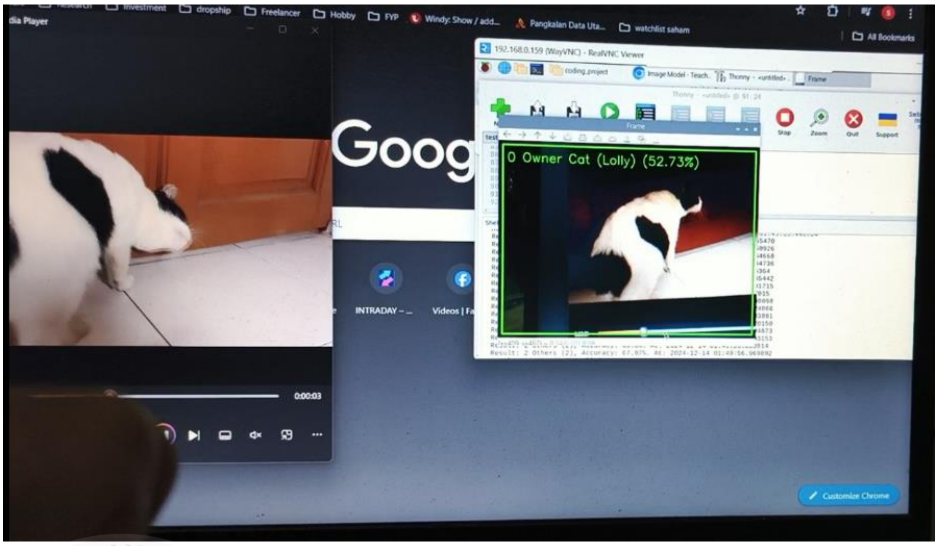


Figure 4.12 Results of using Camera Pi

The figure 4.12 demonstrates a real-time system where the Raspberry Pi camera detects the presence of the owner's cat using a machine learning model trained with Teachable Machine. Once the cat is identified as the intended recipient, the detection triggers a signal to the Raspberry Pi, which then activates a servo motor to dispense food. This system integrates computer vision, machine learning, and automation to create an intelligent and efficient feeding mechanism for the cat.

4.4.3.1 Cat Detection Performance Analysis with Camera Pi and Teachable Machine

Figure 4.13 show the analyse the performance of the detection system, evaluate the accuracy of the machine learning model, and identify potential areas for improvement. By leveraging a three-class classification approach, the system ensures targeted responses, such as feeding the owner's cat while preventing stray cats or other entities from triggering the feeder mechanism. This initiative harnesses the power of affordable computing tools like Teachable Machine to create a model that distinguishes between owner cat, stray cats, and other objects.

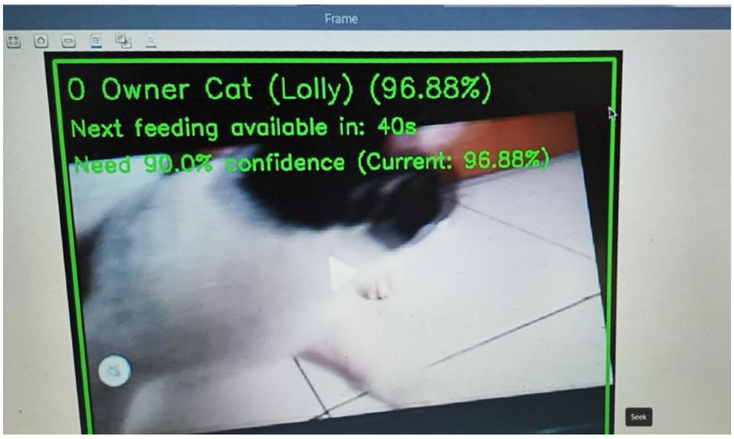


Figure 4.13 Results of Camera Pi Performance

The table 2 and table 3 presents the results of a performance evaluation conducted on the detection model using 20 sample of test run. Each test run represents one of three predefined categories: **Owner Cat**, **Stray Cat**, or **Others**. The purpose of this test is to assess the model's ability to correctly classify the images into their respective categories and measure its confidence in those predictions. By analysing the table, it can analyse how accurately the model identifies the correct category, where it excels, and where improvements are needed.

Sample	Actual Class	Predicted Class	Confidence (%)
1	Owner Cat	Owner Cat	99
2	Owner Cat	Owner Cat	97
3	Owner Cat	Stray Cat	92
4	Stray Cat	Stray Cat	85
5	Stray Cat	Stray Cat	83
6	Stray Cat	Others	87
7	Others	Others	95
8	Others	Others	98
9	Others	Others	96
10	Owner Cat	Owner Cat	93
11	Owner Cat	Stray Cat	89
12	Stray Cat	Stray Cat	82
13	Stray Cat	Others	80
14	Others	Others	97
15	Others	Others	92
16	Owner Cat	Owner Cat	95
17	Stray Cat	Stray Cat	88
18	Others	Others	96
19	Owner Cat	Owner Cat	94
20	Stray Cat	Stray Cat	85

Table 4.2 Cat Classification Results

Actual\Predicted	Owner Cat	Stray Cat	Others
Owner Cat	6	0	0
Stray Cat	1	5	0
Others	0	2	6

Table 4.3 Confusion Matrix for Cat Classification

Prior to creating the graphs, relevant calculations were performed to derive key performance metrics such as Accuracy, Precision, Recall, and F1-Score for each class. These calculations provide a quantitative basis for the visual representations, ensuring the charts accurately reflect the model's performance.

To evaluate the model's performance, the following calculations were performed using the confusion matrix:

1. Recall
2. Precision
3. F1-Score
4. Accuracy

Metric	Owner Cat (%)	Stray Cat (%)	Others (%)	Overall (%)
Recall	85.7	71.4	100	-
Precision	100	83.3	75	-
F1-Score	92.3	76.9	85.7	-
Accuracy	-	-	-	85

Table 4.4 Classification Metrics Summary

Figure 4.14 The Graph of Performance Metrics by Class

The bar chart shows the model's performance metrics for **Owner Cat**, **Stray Cat**, and **Others**. The model performs well for **Owner Cat** (85.7%) and excels in identifying **Others** (100%). However, lower metrics for **Stray Cat** (Recall: 71.4%, F1-Score: 76.9%) highlight challenges in accurately classifying stray cats.

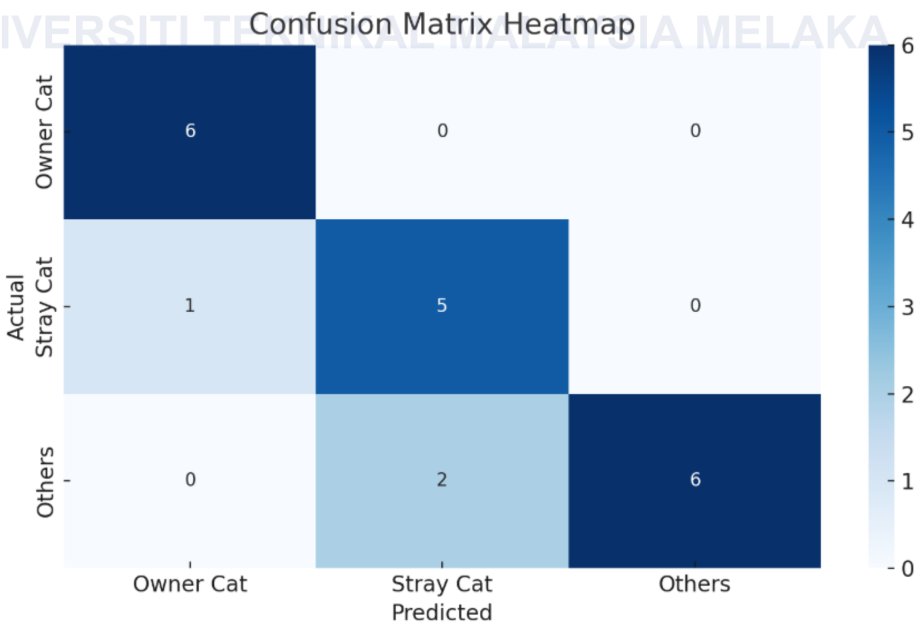


Figure 4.15 The Graph of Confusion Matrix Heatmap

It is a table that summarizes the performance of a classification model by showing the relationship between the actual labels and the predicted labels for a dataset. It provides insights into both correct predictions and errors.

4.4.3.2 Detection Time Comparison in Cat Classification: Lighting and Distance Effects

This analysis explores the impact of lighting conditions and distance on the detection time in a cat classification system, utilizing a Teachable Machine model integrated with a Raspberry Pi Camera to evaluate how these environmental factors influence the system's efficiency and responsiveness.

Condition	Image Capture Time (s)	Processing Time (s)	Total Detection Time (s)
Bright Light, Far	0.50	1.08	1.59
Bright Light, Close	0.99	1.17	2.16
Bright Light, Far	0.87	1.86	2.74
Low Light, Close	1.13	2.27	3.40
Bright Light, Far	0.50	1.42	1.92
Low Light, Far	1.06	1.79	2.86
Low Light, Close	0.87	1.32	2.20
Bright Light, Far	0.54	1.01	1.55
Low Light, Close	1.13	2.25	3.39
Bright Light, Close	1.17	2.32	3.50
Bright Light, Far	0.89	2.35	3.25
Bright Light, Far	0.78	2.43	3.21
Low Light, Close	0.88	2.04	2.93
Bright Light, Far	0.66	2.03	2.69
Bright Light, Close	0.91	1.61	2.53
Bright Light, Far	0.81	1.46	2.28
Bright Light, Close	0.67	2.16	2.84
Bright Light, Close	1.02	2.15	3.17
Bright Light, Far	0.90	1.28	2.18
Low Light, Far	0.96	2.10	3.06

Table 4.5 Detection System Performance Under Varying Conditions

This table 4.5 outlines the performance metrics of a detection system operating under various lighting conditions and distances. It examines how the environment, specifically bright or low light and whether the object is far or close, affects the system's efficiency. The table provides three key metrics: the time taken to capture an image, the processing time needed to analyze the captured image, and the total detection time, which is the sum of the capture and processing times. These metrics help assess the system's overall responsiveness and effectiveness in different scenarios, offering insights into its adaptability to environmental challenges.

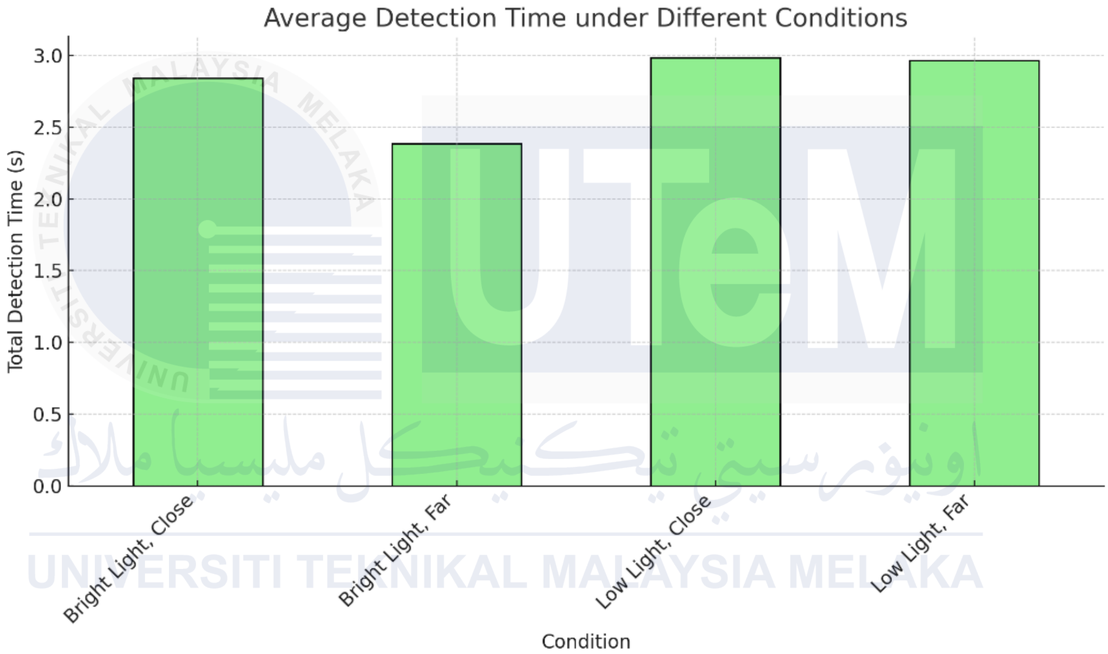


Figure 4.16 The Graph of Average Detection Time under Different Conditions

The table summarizes the detection times under four conditions: **Bright Light, Close**, **Bright Light, Far**, **Low Light, Close**, and **Low Light, Far**. The results show that detection times are shortest under **Bright Light, Far** conditions, averaging around **1.9 seconds**, and longest under **Low Light, Far**, with an average of **3.4 seconds**. This demonstrates that bright lighting significantly improves detection speed, while low lighting and increased distance lead to slower detection times due to challenges in image capture and processing. For consistent and reliable performance, the system should operate in bright lighting, and enhancements such as additional lighting or improved image processing algorithms are recommended for low-light environments.

4.5 Conclusion

The data analysis confirms that the system is highly functional and efficient in automating cat care tasks, with strengths in food dispensing precision and general detection performance. However, the detection system can be further enhanced by addressing challenges in misclassification and improving response times under adverse conditions. These insights provide a clear pathway for system optimization and future developments.

CHAPTER 5

CONCLUSION

5.1 Introduction

This chapter summarizes the findings and achievements of the automated pet feeder project. It reflects on the system's design, functionality, and effectiveness in addressing the challenges faced by pet owners. Additionally, it offers insights into potential improvements and future developments to further enhance the system's reliability and usability.

5.2 Future Recommendation

Future enhancements to the Raspberry Pi-Integrated Automated Pet Feeding System could focus on expanding its versatility and reliability. One potential improvement is the addition of multi-pet support, allowing the system to cater to households with multiple pets by offering individualized feeding schedules and portions. Integrating a battery backup would ensure uninterrupted operation during power outages, providing added convenience and reliability for pet owners.

Using a high-powered Raspberry Pi for a simple pet feeding system may not be the most cost-effective solution. A more lightweight and economical alternative, such as the Raspberry Pi Zero 2 W, is a better fit for small-scale projects. It provides adequate processing power at a fraction of the cost of models like the Raspberry Pi 4 Model B. Additionally, the Raspberry Pi Zero 2 W retains compatibility with various components, making it an ideal choice for a budget-friendly and efficient automated pet feeding system.

The analysis of the automated pet feeder system highlights several key results. The prototype's mechanical design successfully incorporated a food container, funnel, and servo-actuated feeder door, ensuring efficient food dispensing. The software implementation with Python enabled seamless integration of the Raspberry Pi, infrared sensor, servo motor, and Pi camera, allowing for precise detection of food levels and cat recognition. The system's performance in dispensing food was consistent, with each one-second cycle dispensing approximately 30 grams of food, meeting the daily dietary requirement of 200 grams for a 4 kg cat. The detection system demonstrated a high overall accuracy of 85%, with strong performance in identifying the owner's cat and objects classified as "Others". However, lower metrics for stray cat detection indicate areas for improvement.

The evaluation of detection times under varying lighting and distance conditions revealed that the system performed best under bright light and far distances with average 1.9 seconds, while low light and far distances resulted in the longest detection times with average 3.4 seconds. These findings underscore the importance of optimal lighting conditions for efficient operation, with recommendations for additional lighting or enhanced image processing to improve performance in low-light scenarios. Overall, the system effectively automates pet feeding while maintaining precision and adaptability in real-world conditions.

5.3 Conclusion

The Raspberry Pi-Integrated Automated Pet Feeding System successfully automates the pet feeding process, ensuring consistent and precise food dispensing at scheduled intervals. By integrating key hardware components, such as an infrared sensor and servo motor, the system eliminates the need for manual feeding and minimizes food wastage. The project demonstrates the feasibility of using simple technologies to address practical challenges in pet care, particularly for busy pet owners. The system's reliable performance and straightforward operation confirm its potential to simplify pet care routines and provide peace of mind to pet owners.

REFERENCES

[1] Wicaksono, M.A., Subekti, L.B. and Bandung, Y. (2019) 'Development of cat care system based on internet of things', *2019 International Conference on Electrical Engineering and Informatics (ICEEI)* [Preprint]. doi:10.1109/iceei47359.2019.8988788.

[2] *A closer look at pet ownership in Malaysia* (2024) *Standard Insights*. Available at: https://standard-insights.com/blog/pet-ownership-in-malaysia/ (Accessed: 06 June 2024).

[3] Wikipedia contributors. (2024, April 29). Sustainable Development Goal 12. In *Wikipedia, The Free Encyclopedia*. Retrieved 14:16, June 6, 2024, from https://en.wikipedia.org/w/index.php?title=Sustainable_Development_Goal_12&oldid=1221291120

[4] Editor, P. (2018) *Pet hotel: The comparison between commercial and home boarding*, *PetBacker Blog*. Available at: https://www.petbacker.com/blog/facts/pet-hotel-the-comparison-between-commercial-and-home-boarding (Accessed: 06 June 2024).

[5] Editorial, P.E. (2013) *The truth about pet-friendly hotels*, *PetMD*. Available at: https://www.petmd.com/dog/care/the_truth_about_pet_friendly_hotels (Accessed: 06 June 2024).

[6] Patel, T. (2022) *5 reasons you should not send your dog to a Kennel*, *Doglando*. Available at: https://doglando.com/blog/5-reasons-you-should-not-send-your-dog-to-a-kennel/ (Accessed: 06 June 2024).

[7] Lee, J.-J. and Kim, D.-H. (2021) 'Implementation of a smart IOT system with Automatic Pet Feeder', *Journal of Digital Contents Society*, 22(2), pp. 209–214. doi:10.9728/dcs.2021.22.2.209.

[8] J Aparna, P., S, A. and Kusumasree (2023) 'IOT based animal health monitoring system using Raspberry Pi', *International Journal of Advanced Research in Science, Communication and Technology*, pp. 300–307. doi:10.48175/ijarsct-9509.

[9] Carbo, K. (2024) *7 tips on how to feed a cat*, *Purina*. Available at: https://www.purina.com/articles/cat/feeding/guides/how-to-feed-a-cat (Accessed: 06 June 2024).

[10] Overall, K. L., Rodan, I., Beaver, B. V., Carney, H., Crowell-Davis, S., Hird, N., Kudrak, S., Wexler-Mitchel, E., (2005) Panel on Feline Behavior Guidelines, & American Association of Feline Practitioners. Feline behavior guidelines from the American Association of Feline Practitioners. *Journal of the American Veterinary Medical Association*, *227*(1), 70–84. https://doi.org/10.2460/javma.2005.227.70

[11] Kulaikar, J., Kurade, D., Sawant, A., Sthawarmath, P., & Chaurasia, A. (2023). IoT Based Automatic Pet Feeding and Monitoring System. *International Journal of Modern Developments in Engineering and Science*, *2*(4), 24-27

[12] Koley, Soumallya & Srimani, Sneha & Nandy, Debanjana & Pal, Pratik & Biswas, Samriddha & Sarkar, Dr. (2021). Smart Pet Feeder. Journal of Physics: Conference Series. 1797. 012018. 10.1088/1742-6596/1797/1/012018.

[13] Vargas, H. (2020) 'Pet food dispenser design using Raspberry Pi', *International Journal of Emerging Trends in Engineering Research*, 8(4), pp. 1251–1254. doi:10.30534/ijeter/2020/50842020.

[14] Scott, J. (2023) *How much should I feed my cat? tips, advice & faqs*, *RAWZ*. Available at: https://rawznaturalpetfood.com/how-much-should-i-feed-my-cat/ (Accessed: 06 June 2024).

[15] Turner, Dr.B. (2023) *How much and how often you should feed your cat*, *Preventive Vet*. Available at: https://www.preventivevet.com/cats/how-much-and-how-often-to-feed-your-cat (Accessed: 06 June 2024).

[16] Bachman, Z. (2024) *How to determine the right amount of food for your cat: Zooplus magazine*, *zooplus Magazinee*. Available at: https://www.zooplus.co.uk/magazine/cat/cat-nutrition/how-to-determine-the-right-amount-of-food-for-your-cat (Accessed: 06 June 2024).

[17] Dr. Jennifer Coates, D. (2023) *How much to feed a cat: Portions, counting calories & beyond*, *BeChewy*. Available at: https://be.chewy.com/how-much-should-i-feed-my-cat/ (Accessed: 06 June 2024).

[18] Anggraini, N. *et al.* (2020) 'Mobile-based monitoring system for an automatic cat feeder using Raspberry Pi', *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 18(2), p. 1038. doi:10.12928/telkomnika. v18i2.14819.

[19] Vrishanka, P.N. *et al.* (2021) 'Automated pet feeder using IOT', *2021 IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC)* [Preprint]. doi:10.1109/icmnwc52512.2021.9688391.

[20] Wang, R. (2019) 'Design of Mini Pets Feeding Intelligent Home system based on IOT', *Advances in Intelligent Information Hiding and Multimedia Signal Processing*, pp. 31–40. doi:10.1007/978-981-13-9714-1_4.

[21] Nguyen, H. (1970) *Design and implementation of a pet care system*, *Theseus*. Available at: https://urn.fi/URN:NBN:fi:amk-202004024348 (Accessed: 07 June 2024).

[22] Fadhilah, D. (2013) *Kapan Dan Berapa Banyak Kucing Harus Makan – berbagi informasi Mengenai Ilmu veteriner*, *https://ilmuveteriner.com/kapan-dan-berapa-banyak-kucing-harus-makan/*. Available at: https://ilmuveteriner.com/kapan-dan-berapa-banyak-kucing-harus-makan/amp/ (Accessed: 06 June 2024).

[23] Dr. Jennifer Coates, D. (2023) *How much to feed a cat: Portions, counting calories & beyond*, *BeChewy*. Available at: https://be.chewy.com/how-much-should-i-feed-my-cat/ (Accessed: 07 June 2024).

[24] Sunil K., Vishwanath S., Vikas T., Avinash K., Jagadish J.,"IOT Based Dog Day-care Robot,"International Research Journal of Modernization in Engineering Technology and Science, vol. 4, Issue 7, pp. 2617-2620, Jul. 2022.

[25] Quiñonez, Y. *et al.* (2021) *Communication architecture based on IOT technology to control and monitor pets feeding, JUCS*. Available at: https://lib.jucs.org/article/65094/ (Accessed: 07 June 2024).

[26] Vargas, H. (2020) 'Pet food dispenser design using Raspberry Pi', *International Journal of Emerging Trends in Engineering Research*, 8(4), pp. 1251–1254. doi:10.30534/ijeter/2020/50842020.

[27] Kim, S. (2016) 'Smart pet care system using internet of things', *International Journal of Smart Home*, 10(3), pp. 211–218. doi:10.14257/ijsh.2016.10.3.21.

[28] Marlina, E. (2023) 'Design of smart prototype pet feeder using passive infared (PIR) sensors', *Journal of Computer Networks, Architecture and High Performance Computing*, 5(1), pp. 303–310. doi:10.47709/cnahpc.v5i1.2237.

[29] Lee, J.-J. and Kim, D.-H. (2021) 'Implementation of a smart IOT system with Automatic Pet Feeder', *Journal of Digital Contents Society*, 22(2), pp. 209–214. doi:10.9728/dcs.2021.22.2.209.

[30] Saswadkar, V., Paygude, V. and Gaikwad, P.D. (2018) 'Pet care system based on Android application', *International Journal of Trend in Scientific Research and Development*, Volume-2(Issue-4), pp. 1488–1491. doi:10.31142/ijtsrd14415.

[31] Eun-Mo Ahn, Dong-Hoi Kim, "Implementation of Integrated Platform of Face Recognition CCTV and Home IoT(Internet of Things)", The Journal of Digital Contents Society, Vol. 19, No. 2, pp. 393-399, Oct. 2018

[32] Bureau, E. (2023). *Teaching a computer to recognise cats*. Electronics For You. Available at https://www.electronicsforu.com/electronics-projects/teaching-computer-recognise-cats (Accessed: 10 June 2024).

APPENDIX

A. Phyton Programming Raspberry Pi Camera & Servo Motor & Infrared Sensor

Phyton Programming Raspberry Pi Camera & Servo Motor & Infrared Sensor

```
from tflite_runtime.interpreter import Interpreter
import numpy as np
from picamera2 import Picamera2
import cv2
import datetime
import RPi.GPIO as GPIO
from gpiozero import AngularServo
from gpiozero.pins.pigpio import PiGPIOFactory
from time import sleep, time

# Pin and GPIO setup
IR_PIN = 23
SERVO_PIN = 18
GPIO.setmode(GPIO.BCM)
GPIO.setup(IR_PIN, GPIO.IN)

# Servo setup
factory = PiGPIOFactory()
servo = AngularServo(SERVO_PIN, min_angle=0, max_angle=180,
                     min_pulse_width=0.0005,
max_pulse_width=0.0024,
                     pin_factory=factory)
# Set default position
servo.angle = 0


# Load labels from file
def load_labels(path):
    with open(path, 'r') as f:
```

```
        return [line.strip() for line in f.readlines()]


# Set input tensor for the TFLite model
def set_input_tensor(interpreter, image):
    tensor_index = interpreter.get_input_details()[0]['index']
    input_tensor = interpreter.tensor(tensor_index)()[0]
    input_tensor[:, :] = image


# Classify image and return the top result
def classify_image(interpreter, image, top_k=1):
    set_input_tensor(interpreter, image)
    interpreter.invoke()
    output_details = interpreter.get_output_details()[0]
    output = 
np.squeeze(interpreter.get_tensor(output_details['index']))
    scale, zero_point = output_details['quantization']
    output = scale * (output - zero_point)
    ordered = np.argpartition(-output, 1)
    return [(i, output[i]) for i in ordered[:top_k]][0]


# Check if food is present using IR sensor
def check_food_level():
    return GPIO.input(IR_PIN) == GPIO.LOW  # True if food is 
present


# Dispense food
def dispense_food():
    servo.angle = 90  # Open dispenser
    sleep(1)        # Keep open for 1 second
    servo.angle = 0  # Close dispenser
```

```
# Paths to model and labels
data_folder = "/home/pi/coding_project/"
model_path = data_folder + "model.tflite"
label_path = data_folder + "labels.txt"

# Load the model and allocate tensors
interpreter = Interpreter(model_path)
interpreter.allocate_tensors()

# Get model input shape
input_shape = interpreter.get_input_details()[0]['shape']
height = input_shape[1]
width = input_shape[2]

# Load labels
labels = load_labels(label_path)

# Initialize Picamera2
picam2 = Picamera2()
config = picam2.create_preview_configuration(main={"size": (640,
480), "format": "RGB888"})
picam2.configure(config)
picam2.start()

# Track last feeding time
last_feeding_time = 0
FEEDING_COOLDOWN = 60 #300   5 minutes in seconds
CONFIDENCE_THRESHOLD = 0.90  # 90% confidence threshold

try:
```

```
    while True:
        # Capture and process frame
        frame = picam2.capture_array()
        image = cv2.resize(frame, (width, height))

        # Perform classification
        label_id, prob = classify_image(interpreter, image)
        classification_label = labels[label_id]
        current_time = time()

        # Check if Lolly is detected with high confidence
        if "Owner Cat (Lolly)" in classification_label and prob
>= CONFIDENCE_THRESHOLD:
            color = (0, 255, 0)  # Green

            # Check if enough time has passed since last feeding
            if current_time - last_feeding_time >=
FEEDING_COOLDOWN:
                # Check if food bowl is empty
                if not check_food_level():
                    print(f"Dispensing food for Lolly
(Confidence: {prob*100:.2f}%)")
                    dispense_food()
                    last_feeding_time = current_time
        else:
            color = (0, 0, 255)  # Red

        # Draw rectangle and add text to frame
        cv2.rectangle(frame, (10, 10), (630, 470), color, 4)
        font = cv2.FONT_HERSHEY_SIMPLEX
        text = f"{classification_label} ({np.round(prob * 100,
```

```
        cv2.putText(frame, text, (20, 50), font, 1, color, 2)


        # Add confidence threshold indicator
        if "Owner Cat (Lolly)" in classification_label:
            confidence_text = f"Need {CONFIDENCE_THRESHOLD*100}%
confidence (Current: {prob*100:.2f}%)"
            cv2.putText(frame, confidence_text, (20, 130), font,
0.8, color, 2)


        # Add time until next feeding
        time_remaining = max(0, FEEDING_COOLDOWN - (current_time
- last_feeding_time))
        next_feed_text = f"Next feeding available in:
{int(time_remaining)}s"
        cv2.putText(frame, next_feed_text, (20, 90), font, 0.8,
color, 2)


        # Display the frame
        cv2.imshow("Frame", frame)


        # Exit on 'q' key press
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
except KeyboardInterrupt:
    print("\nProgram stopped by user")
finally:
    # Cleanup
    picam2.stop()
    cv2.destroyAllWindows()
    GPIO.cleanup()
    servo.angle = 0  # Return to default position
```