DEVELOPMENT OF AN AUTOMATED SELF-BILLING COUNTER WITH BARCODE SCANNING AND IMAGE PROCESSING ON RASPBERRY PI

NURUL HAZREEN BINTI KHAIRUL SALEH

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEVELOPMENT OF AN AUTOMATED SELF-BILLING COUNTER WITH BARCODE SCANNING AND IMAGE PROCESSING ON RASPBERRY PI

NURUL HAZREEN BINTI KHAIRUL SALEH

This report is submitted in partial fulfillment of the requirements for the degree of Bachelor of Electronics Engineering Technology (Telecommunication) 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 Self-Billing Counter with Barcode Scanning and Image processing on Raspberry Pi

Sesi Pengajian : 2024/2025

Saya NURUL HAZREEN BINTI KHAIRUL SALEH 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:

(TANDATANGAN PENULIS) (COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: ....

TS. DR. IDA SYAFIZA BINTI MD ISA
Pensyarah Kanan
Fakulti Teknologi dan Kejuruteraan Elektronik dan Komputer
Universiti Teknikal Malaysia Melaka

Tarikh : 24 January 2025 Tarikh : 24 January 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 Self-Billing Counter with Barcode Scanning and Image Processing on 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	:	NURUL HAZREEN BINTI KHAIRUL SALEH
Date	:	24 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 Electrical Engineering Technology with Honours.

Signature :

Supervisor Name : TS. DR. IDA SYAFIZA BINTI MD ISA

Date : 24 JANUARY 2025

DEDICATION

To

My father, Zainal Azzudin Bin Nordin for his love, encouragement, and patience.

My mother, Suryana Binti Rosmani for her love, support, and sacrifice.
She continues to be a source of inspiration to me throughout my life

My siblings, Hazrik Addin, Hairil Adam, and Zafran Aiman are always there to offer a moment of clarity.

&

I must give special thanks to my supervisor Ts. Dr. Ida Syafiza binti Md Isa and my close friends for their support and unconditional guidance, without which 1 would never have completed this project.

ABSTRACT

Nowadays, self-billing systems have gained popularity in retail environments. This is because it offers convenience and efficiency to customers. However, there are a few challenges with the self-billing systems. This includes theft, inaccurate item identification, and slow processing times. Therefore, in this work, an automated self-billing counter that integrates image processing and a barcode reading technology on Raspberry Pi has been developed. The primary focus of this work is to enhance the self-checkout system in businesses such as small supermarkets or large hypermarkets while reducing the required number of cashier machines. The system leverages image recognition and processing techniques to detect the products before they are weighed. Teachable Machine, TensorFlow, and OpenCV were employed for image classification, while the hx711 library was used for weight measurement to verify the type of item. The performance of the developed automatic self-billing system is evaluated by considering nine different products where five products through barcode scanning and the remaining four products are through image processing. Additionally, customer purchase history and total bills are logged into Google Drive for management purposes, and receipts are sent to customers via WhatsApp using Twilio. By implementing these technologies in the automatic self-billing system, the customer will gain a good quality of services such as an efficient and seamless purchasing process. Several situations were investigated, including product detection under various lighting circumstances, item recognition from different angles, and weight measures to ensure data consistency. The results show the accuracy for each product class exceeds 70%, indicating the possibility of a more reliable and secure self-billing system.

ABSTRAK

Kini, sistem pengiraan sendiri telah menjadi popular dalam persekitaran runcit. Ini kerana ia menawarkan kemudahan dan kecekapan kepada pelanggan. Namun, terdapat beberapa cabaran dengan sistem pengisian sendiri. Ini termasuk kecurian, pengenalan item yang tidak tepat, dan masa pemprosesan yang perlahan. Oleh itu, dalam kerja ini, kaunter pengisian diri automatik yang mengintegrasikan pemprosesan imej dan teknologi pembacaan kod bar pada Raspberry Pi telah dibangunkan. Fokus utama kerja ini adalah untuk meningkatkan sistem pembayaran sendiri di perniagaan seperti pasar raya kecil atau pasar raya besar sambil mengurangkan bilangan mesin juruwang yang diperlukan. Sistem ini memanfaatkan teknik pengenalan dan pemprosesan imej untuk mengesan produk sebelum ia ditimbang. Teachable Machine, TensorFlow, dan OpenCV digunakan untuk pengelasan imej, manakala perpustakaan hx711 digunakan untuk pengukuran berat bagi mengesahkan jenis item. Prestasi sistem pengiraan diri automatik yang dibangunkan dinilai dengan mempertimbangkan sembilan produk yang berbeza di mana lima produk melalui pengimbasan kod bar dan empat produk yang selebihnya melalui pemprosesan imej. Selain itu, sejarah pembelian pelanggan dan jumlah bil dicatat dalam Google Drive untuk tujuan pengurusan, dan resit dihantar kepada pelanggan melalui WhatsApp menggunakan Twilio. Dengan melaksanakan teknologi-teknologi ini dalam sistem pengebilan automatik sendiri, pelanggan akan memperoleh kualiti perkhidmatan yang baik seperti proses pembelian yang cekap dan lancar. Beberapa situasi telah disiasat, termasuk pengesanan produk di bawah pelbagai keadaan pencahayaan, pengecaman item dari sudut yang berbeza, dan ukuran berat untuk memastikan konsistensi data. Keputusan menunjukkan ketepatan untuk setiap kelas produk melebihi 70%, menunjukkan kemungkinan sistem pengisian diri yang lebih boleh dipercayai dan selamat.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my gratitude to my supervisor, Dr. Ida Syafiza Binit Md Isa for the precious guidance, words of wisdom, and patience throughout this project.

I am also indebted to Universiti Teknikal Malaysia Melaka (UTeM) for the financial support that enabled me to accomplish the project. Not forgetting my colleagues, Farzana Syafiah, Armyza Sharina, Nur Insyirah Izaati, Nurul Fatihah, and Adi Waris for their willingness to share their thoughts and ideas regarding the project.

My highest appreciation goes to my parents, and family members for their love and prayer during the period of my study.

Finally, I would like to thank all the staff at the laboratory, colleagues and classmates, the faculty members, and other individuals who are not listed here for being cooperative and helpful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

kg	-	Kilogram
	-	
	-	
	-	
	-	
	-	
	-	
	-	

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

In recent years, hypermarkets and supermarkets have grown in popularity across Malaysia offering convenience shopping for a range of daily necessities and services. Also, the most popular local hypermarkets such as Mydin, NSK Trade City, and Econsave including foreign hypermarkets such as AEON BiG, and Lotus have established a strong presence in the Malaysian market.

Shopping at hypermarkets or supermarkets can be difficult and time-consuming for customers because of long queues and delays at the checkout counter. Hence, the increasing number of customers and the low manpower at the checkout counter will increase the difficulties of managing the transactions at the checkout counter. It will become worse, especially during the weekends, holidays, and special occasions due to the prolonged wait times and adding frustration among the shoppers[1].

The project in [2] designed a smart billing and direction-controlled trolley that automatically adds up the prices of items placed in the trolley using RFID tags and sends the total bill to the customer's mobile app via Bluetooth. There is also a risk of theft if items are not properly scanned. Besides that, a smart trolley with an automatic billing system [3] follows the customer using a web camera to recognize a specific tag and calculates the total bill of items added using RFID tags. While it reduces the need to push the trolley and eliminates long billing queues, it might struggle to navigate in crowded areas or accurately identify the customer.

The development of an automated self-billing counter utilizing Raspberry Pi has been presented as a solution to the problem of technological convergence. Using a barcode scanning and image processing method, the system can detect products entered into the self-checkout counter with a barcode labeled at the product and weigh the products based on their category class using image classification. An automated self-billing counter sends the receipt bill to the customer through WhatsApp and the purchase history is recorded in Google Drive for management purposes. The automated self-billing counter will benefit customers and businesses by minimizing the checkout queues, thereby reducing the waiting time.

Traditionally, as mentioned above, grocery or other retail shopping has been a manual and time-consuming process as it, often involves a long queue at the checkout counter due to, manual item scanning, besides having a high possibility of human errors in billing [4]. Therefore, the proposed automated self-billing counter using Raspberry Pi with a barcode scanner and image recognition and processing system can overcome the delay at the checkout counter, as well as minimize human error in the billing process.

1.2 Addressing Societal/Global Issues Through Development of Automated of Self–Billing Counter with Barcode Scanning and Image Processing with Raspberry Pi

There are several societal/global issues found at checkout counters which are long waiting times, theft, and retail fraud. Long wait times at checkout counters [5] can be inconvenient for customers, particularly for those with mobility issues, the elderly, or parents with children. Waiting in line can be physically challenging for individuals with limited mobility, causing discomfort and potentially exacerbating existing health conditions. For the elderly, long wait times can be exhausting and may lead to feelings of frustration. Parents with children also may be juggling multiple responsibilities to keep their children under control. The automated self-billing counter can help reduce wait times by improving efficiency and enhancing the overall shopping experience for customers of all ages and abilities.

Secondly, theft and retail fraud are pervasive issues that impact businesses across the globe [6]. These crimes involve a variety of tactics, including shoplifting, employee theft, and fraudulent transactions, leading to significant financial losses for retailers. One popular type of retail theft is shoplifting, in which customers steal products from stores without paying. Employee theft occurs when staff members steal products or money from their employer. This could harm the customer's trustworthiness and the reputation of the company. The system can solve the issues by accurately tracking inventory and ensuring all items are properly scanned and paid for.

In conclusion, the automated self-billing counter system represents a significant advancement in addressing societal and global challenges, particularly with customer convenience and trust in the retail sector. This technology can help minimize paper waste by delivering receipt invoices to customers electronically. This technology will enhance the customer's purchasing experience by making it more efficient and inclusive, hence fostering a more joyful environment.

1.3 Problem Statement

The design of the traditional checkout counter raised several challenges for both retailers and consumers. Some of these issues are long queue times as the operations are performed manually besides introducing high human error in the billing process. Due to this, an effective and appropriate solution is needed to overcome the problem to enhance consumer satisfaction levels as well as meet the needs of the community in the retail context.

Long queues are the main problem in modern supermarkets, which lead to long waiting times, thus reducing the quality of service to the consumers. When the number of consumers increases especially during peak hours or holidays [1], the problem will become worse which puts pressure on the employees to meet the quality of service to the consumers.

In addition to this, most hypermarkets use full-time or part-time employees to fulfill the staffing requirements of the store. Some of the challenges that retailers may come across may include inaccurate prices, failure in scanning, and other mistakes since the staff undergoes a limited amount of training. As a result, the retailer is solely liable for human errors [4].

Most smart cart systems [1], [7], [8] utilize RFID technology as an innovation to the system. The application of RFID technology is indubitably one of the most useful competitive tools for establishment in the global market. But at the end of the logistic chain where the end consumer is located, RFID tags typically end up in the form of waste [9].

Traditional cashier counters face various issues dealing with mostly theft and retail fraud, which proves to be very disturbing to the retailers. Customers are said to shoplift when they put items in their clothes, purses, or bags when they pay or walk out without paying. In addition, the cashier may also take the money from the register or give unauthorized discounts.

Customers also exploit price and labeling mistakes to obtain items for less money. Return fraud is also represented by cases when someone returns stolen products for compensation. Retailers are thus forced to invest more untowardly in security measures and preventive steps that lower the profitability of stores.

To solve these problems, automated self-billing counters with a barcode and image recognition and processing system are essential to reduce waiting times and minimize errors thus improving the retail experience for customers while driving operational efficiency for businesses.

1.4 Project Objective

a) To develop an automated self-billing counter with a barcode scanning and image recognition and processing system using Raspberry Pi.

b) To develop a dashboard for an automated self-billing counter system using the Google Drive application.

c) To evaluate the performance of the developed system in terms of its accuracy and reliability.

1.5 Scope of Project

a) The automated self-billing counter system prototype will be tested with 9 products.

b) The prototype will be evaluated in terms of the accuracy for 50 repeated items consisting of 4 products that require load sensors with an image processing technique and 5 products using a barcode scanning technique.

1.6 Conclusion

In conclusion, the development of the automated self-billing counter using Raspberry Pi, barcode scanning, and image processing offers a viable solution to the common issues faced at traditional checkout counters, such as long wait times, billing inaccuracies, and retail fraud. By implementing this system, the project aims to streamline the checkout process, minimize human errors, and enhance customer satisfaction. The integration of both barcode scanning and image processing technologies improves efficiency while ensuring accurate billing, reducing the risk of theft, and optimizing inventory management. The implementation of electronic receipts and secure payment processes not only fosters trust between retailers and customers but also minimizes paper waste, contributing to a more sustainable retail environment. Furthermore, the accuracy and reliability of the system, as demonstrated through prototype testing, show promising potential to streamline operations, optimize resource utilization, and ultimately boost the profitability of retail stores. This project stands as a significant step forward in modernizing retail checkout processes, ensuring efficiency, accuracy, and security in the fast-paced shopping landscape.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter summarizes the findings of several researchers regarding the design and development of an automated self-billing counter. The literature study was created by doing research from many different sources and applying key terms such as “cart system”, “self-checkout counter”, and “smart shopping”.

2.2 Hypermarket

Hypermarket stores are very large retail stores governed by chain contracts with franchise headquarters [10], [11]. According to Mordor Intelligence [12], hypermarket store is projected to be worth USD 774.27 billion in 2024 and increase by USD 876.03 billion by 2029 with a compound annual growth rate (CAGR) of 2.5% from 2024 to 2029.

In addition, hypermarkets also tend to operate for 12 hours to 24 hours a day, seven days a week, thus making them a popular place to buy groceries [11]. People who live in metropolitan cities such as Kuala Lumpur are often busy and have limited time, the hypermarket places they always visit when they have free time.

Hypermarkets are generally located in places that are easily accessible areas for the convenience of customers such as residential areas, or in busy city centers. Hypermarkets have 80,000 to 200,000 square meters and large parking lots. Because of that element, most people such as families, housewives, and workers frequently go to hypermarkets as one-stop shopping that focuses more on fast-moving consumer products.

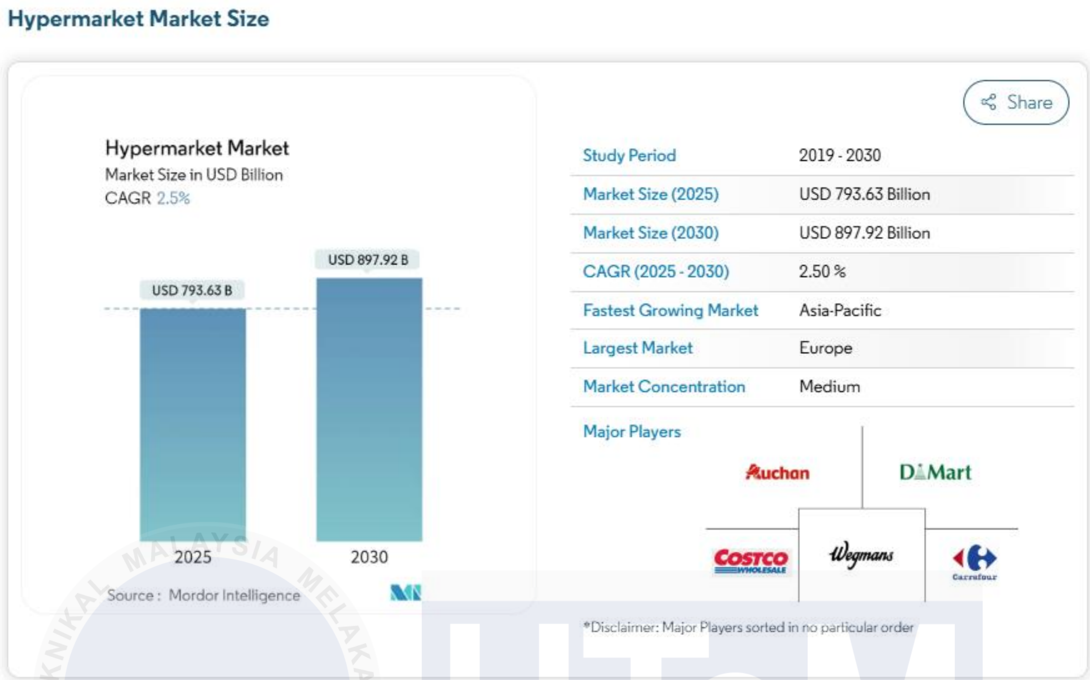


Figure 2. 1 Hypermarket analysis [12]

2.3 Checkout Counter

A checkout counter is the designated area in a retail store where customers pay for their purchases. Checkout counters are typically staffed by cashiers or sales associates who assist customers with completing their transactions, bagging items, and providing customer service. The checkout counter plays a critical role in the overall shopping experience and customer satisfaction.

A study by [13], [14] found that long queues and slow checkout times were the top factors contributing to customer dissatisfaction and abandoned purchases. In response, they implemented various strategies to optimize the checkout counters, such as implementing the self-checkout system [6], [7], [15] and introducing mobile point-of-sale (POS) devices [16].

2.4 Self-Checkout Counter

Nowadays, most stores all over the world have replaced their traditional checkout counter with self-checkout counters, aiming to reduce expenses in human resources and enhance the shopping experience. Recently, the prototypes of the self-checkout system which in machine learning is used to recognize products and aims to satisfy the business' worries by improving the consumers' purchasing experience and minimizing the possibility of robberies [14].

Several self-checkout counter systems have been innovated with a combination of Internet of Things (IoT) advantages to create practical and effective products such as RFID technology [7], [17], [18], deep learning [19], and image detection [17], [19].

2.5 Comparison of Checkout Counter vs Self-checkout Counter

The checkout counter and the self-checkout counter are two prominent systems used in retail environments to facilitate customer transactions. The following comparison highlights the differences between these two systems to understand their impact on efficiency and customer satisfaction.

Table 2. 1 Comparison of Checkout Counter and Self-Checkout Counter

Feature	Checkout counter	Self-checkout counter
Staffing requirement	Need a cashier or staff	Minimal staff needed to supervise
Speed of service	Depends on cashier efficiency	Faster due to technology like RFID
Security concerns	Has the risk of theft	Reduces risk
Error frequency	Higher error	Lower due to accuracy
Space requirement	More space is needed per counter	Less space needed per unit

2.6 Existing Self-Checkout Counter

This section provides an overview and an explanation of the literature review related to the existing self-checkout counter.

2.6.1 NCR SelfServ Checkout

NCR SelfServ Checkout is the first self-service checkout system that Tesco Malaysia launched. It guides the customers to buy products by themselves using a touchscreen throughout the process and reduces the time spent waiting in line [6], [13]. The system requires customers to scan the barcodes one by one themselves [13], [20]. After that, the system will identify the type of item. The customer can continue paying by using a credit or debit card machine. NCR is currently the leading provider of self-checkouts. According to NCR, "self-checkout solutions can reduce wait times by as much as 40% while nearly two-thirds of shoppers say stores that offer the option of self-checkout provide better customer services".

Figure 2. 2 NRC SelfServ Checkout [6]

2.6.2 Smart Cart System

A study in [21] introduced the cart system that utilizes RFID technology, an Arduino module, an RFID reader, an LCD, and a Wi-Fi module to streamline the shopping experience. Customers scan products with RFID tags using the cart's reader, displaying real-time LCD pricing information. Secure data transfer to the Point-of-sale (PoS) server via Wi-Fi enables efficient billing, while an RFID smart card facilitates payment. The smart cart system offers a cost-effective and time-efficient solution to enhance the overall shopping experience for customers.

According to [4], the system's components include an Arduino Atmega2560 microcontroller, an IR sensor, a scanner for barcodes, a power supply, an LCD, a modem for GSM, a solenoid for the door, a swiping machine, and a servo motor. These components collectively participate in offering features, for instance, product identification with the barcode, displaying the count of the items and the total cost on the LCD, transmitting information to the central server and the customer's mobile phone through GSM, unlocking the door through RFID for accessing purchased items, and the swiping machine for payment.

In [22] employed the creative solution in shopping using the arts of using Raspberry Pi, barcode readers, and LCDs, which enabled users to do the self-scanning of the products to generate invoices and payments using the intelligent IP scanner, barcode scanning, and database with IOT. The friendly user interface ensures that all the shopping purchases are well registered; it increases shopping efficiency when used in collaboration with the load cell module that develops real-time transmission of information.

The work [8] incorporates RFID and IoT features to improve the customers' experience. These are the EM-18 RFID module for reading RFID tags, Node MCU for IoT integration, buzzer for alarm, LCD for display, add/delete switch for augmenting/deleting data, and a final switch for command. The proposed system will make different shopping processes more accessible and more effective, enhance stock control, and offer customers a practical and easy shopping experience in the retail setting.

Figure 2. 3 Smart Cart System [13]

2.6.2.1 Comparison of Smart Cart System

The concept of a smart cart system has been explored in various research papers, each proposing unique designs and functionalities. These designs incorporate different technologies, such as RFID, IoT, and microcontrollers, to enhance shopping experiences by improving efficiency. Table 2.2 provides a comparison of features, advantages, and limitations across different smart cart systems.

Table 2. 2 Comparison of Smart Cart System

No.	**Research paper**	Smart Cart: A Distributed Framework [21]	Smart Electronic Trolley for shopping mall [4]	Smart Shopping Trolley Using IoT [22]	RFID-Powered Intelligent Shopping Cart [8]
1.	**Microcontroller**	Arduino	Arduino Atmega 2560	Raspberry Pi	NodeMCU

2.	**RFID technology**	Yes	No	No	Yes
3.	**LCD Display**	Yes	Yes	Yes	Yes
4.	**Wi-Fi module**	Yes	No	Yes (through Raspberry Pi)	Yes (NodeMCU)
5.	**GSM module**	No	Yes	No	No
6.	**Add/delete switch**	No	No	No	Yes
7.	**IoT Integration**	Yes	No	Yes	Yes
8.	**Database Integration**	Yes	No	Yes	No
8.	**Real-time Data Transfer**	Yes	Yes	Yes	Yes
9.	**Alert Mechanism**	No	No	No	Yes
10.	**Advanced technologies**	No	No	Advanced IP scanner, Load Cell Module	No
11.	**Advantages**	Improved security	Time efficiency	Data storage and analysis	High accuracy tracking items
12.	**Disadvantages**	Limited range of RFID	Cost maintenance	Limited compatibility.	Privacy concerns

2.6.3 Automatic Billing Using Deep Learning Technology

The work in [23] introduced to streamline the fruit and vegetable purchasing process by utilizing a Raspberry Pi 3 connected to a load cell, USB camera, and USB printer. The load cell triggers the system by detecting the weight of the fruit, while the USB camera captures an image for classification using a ResNet-152 CNN model. The system calculates the total price based on the fruit type and weight, generates a detailed receipt with barcode and QR code information, and prints it using the USB printer. Unique features include a one-time load cell calibration process and an innovative approach to image capture, ensuring accuracy and efficiency in the billing process.

In [19], the automatic billing system based on instance segmentation is equipped with a camera for image capture, a light source for illumination, a display screen for showing segmented instances and prices, and a panel for placing plates with dishes. The device automates the billing process by capturing images of the plates, processing them using the segmentation model, and displaying the segmented instances with their respective prices in real-time. Additionally, the system features a FoodSyn module for synthesizing images by cropping food parts and pasting them onto plates, an encoder-decoder network for segmenting instances in the generated images, and a smart IoT device for deploying the model for real-time automatic billing. These components work together to streamline the billing process, enhance user experience, and provide accurate pricing for different dishes based on the plates they are served on.

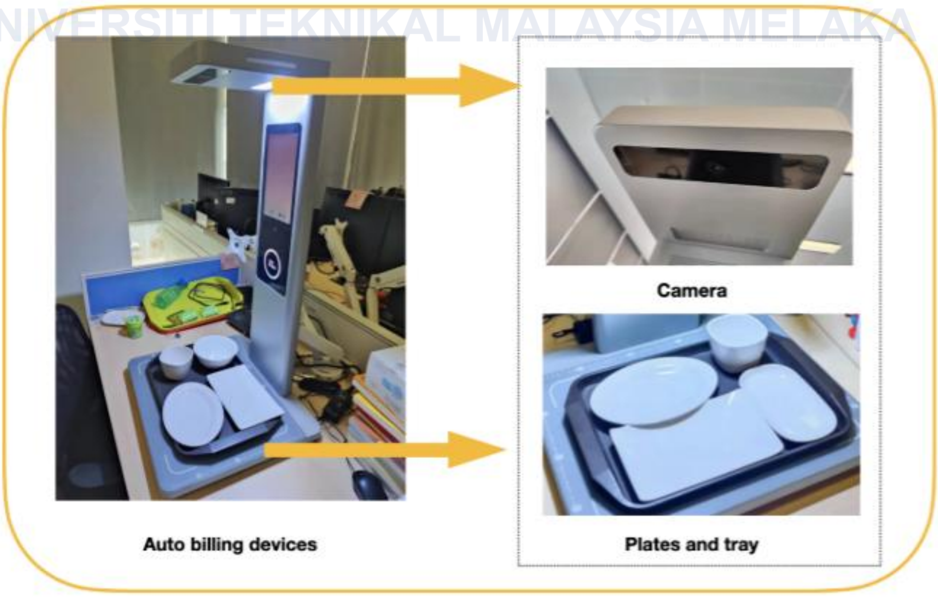


Figure 2. 4 An Automatic Billing System based on Instance Segmentation [19]

2.6.3.1 Comparison of Deep Weight: A Semi-Autonomous Food Weight Billing System Using Deep Learning and A Real-Time Chinese Food Auto Billing System

This section compares two systems, Deep Weight and A Real-Time Chinese Food Auto Billing System, focusing on their method for image processing and classification, advantages, and limitations. These comparisons highlight the potential and challenges of integrating deep learning real-world billing applications.

Table 2. 3 Comparison of the system using Deep Learning

No.	**Research paper**	Deep Weight: A Semi-Autonomous Food Weight Billing System using Deep Learning [23]	A Real-Time Chinese Food Auto Billing System Based on Instance Segmentation [19]
1.	**Processor**	Raspberry Pi 3	Smart IoT Device
2.	**Image capture**	USB Camera	Camera
3.	**Image Classification**	ResNet-152 CNN	Segmentation model with encoder-decoder network
4.	**Billing process**	Based on fruit type and weight	Based on segmentation instances of food on plates
5.	**Unique features**	One-time load cell calibration, image-based fruit classification	Image segmentation, FoodSyn module for synthesizing images
6.	**Total Price Calculation**	Based on the type and weight of the fruit	Based on segmented instances and respective prices
7.	**Advantages**	Quickly generate information and accurately classify items.	It can be deployed on both Android mobile phones and IoT devices
8.	**Disadvantages**	Technical complexity	Potential overfitting

2.6.4 Raspberry Pi-based Shopping Cart Following Robot through Computer Vision

and Object Tracking.

The researcher in [24] proposed a way to lessen the effort of customers, especially in grocery areas or supermarkets. In this research paper, Raspberry Pi 3 B+ is the central processing unit, Arduino UNO for sensor connectivity, including ultrasound and load cell sensors, a barcode reader for product identification, a buzzer for alerts, and an LCD screen for displaying information. Motor drivers control cart movement, while a web camera enables object tracking. Python is the programming language that is used in the system. Through a smooth integration of hardware and software, these parts act together to enhance the entire shopping experience by enabling autonomous following of consumers, obstacle avoidance, weight measurement, product information display, and alert notifications.

Figure 2. 5 Shopping Cart Robot through Computer Vision and Object Tracking [24]

2.7 Comparison of the previous work

This section presents a comparison of the previous studies related to automated self-checkout systems, smart shopping trolleys, and similar technologies. By examining these works, their advantages and disadvantages are identified. This comparison provides a foundation to emphasize the novelty and enhancements introduced by the proposed system, ensuring it addresses the gaps and challenges highlighted in prior research.

Table 2. 4 Comparison of previous work

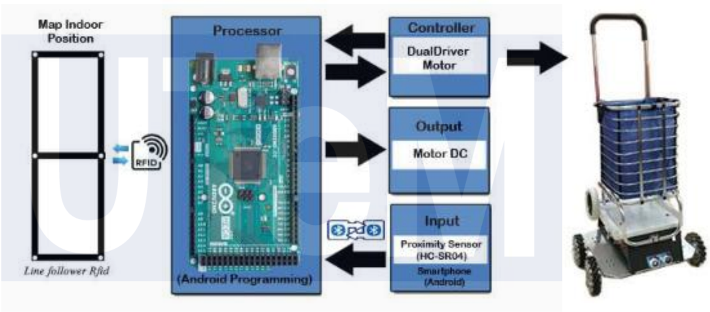
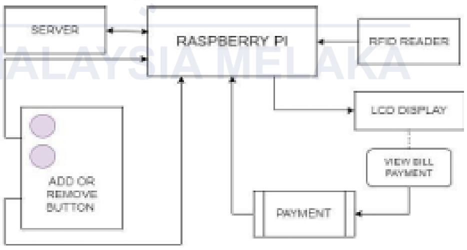
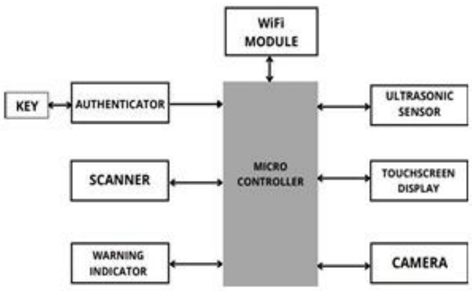
No.	Name of research paper	Advantages	Disadvantages
1.	Line Follower Smart Trolley System V2 using RFID [25]	Efficient item location without manual searching. Reduces reliance on staff for assisting customers.	Limited battery life of the robot. Complexity algorithms to handle real-time navigation.
2.	Autonomous Billing Cart for Retail Stores [18]	Increased efficiency. Better security and control.	Customer adoption barriers. Data privacy concern.
3.	Smart Shopping Trolley System [26]	Time saving. High accuracy. Cost-effective and scalable.	Performance may degrade in busy areas. Potential error in complex scenarios.
4.	IoT Based Smart Billing and Direction controlled Trolley [26]	Real-time price updates. Hands-free shopping because the trolley follows customers automatically.	High cost. Battery dependency.

			Technical reliability of Bluetooth in crowded environments.
5.	Deep Weight: A Semi-Autonomous Food Weight Billing System using Deep Learning [23]	Quickly generate information and accurately classify items.	Complexity.
6.	A Real-Time Chinese Food Auto Billing System Based on Instance Segmentation [19]	Deployed on both Android mobile phones and IoT devices.	Overfitting.
7.	RFID-Powered Intelligent Shopping Cart	Real-time inventory management. Improved efficiency. Scalable and Adaptable.	Battery life and power dependency. RFID systems face signal interference from metals or liquids.
8.	Low-Cost Automatic Purchase Cart [27]	Cost-effective. Fraud prevention. Durability because RFID can be long-lasting and reliable.	The system relies heavily on server availability. Limited range of RFID

2.8 The process flows of the previous work

This section outlines the process flows of existing systems related to automated self-checkout and smart shopping technologies. By reviewing the operational steps and methodologies of previous works, a clear understanding of their system designs, workflows, and implementation strategies can be achieved. This analysis helps to identify key components, bottlenecks, and areas for improvement that can addressed in the proposed system.

Table 2. 5 The process flow of the previous work

No.	Name of research paper	Model/illustration project
1.	Line Follower Smart Trolley System V2 using RFID [25]	 Figure 2. 6 Architecture of Smart Trolley System V2 [25]
2.	Autonomous Billing Cart for Retail Stores [18]	 Figure 2. 7 Block diagram of Autonomous Billing Cart [18]
3.	Smart Shopping Trolley System [26]	 Figure 2. 8 Block diagram of Smart Shopping Trolley[26]

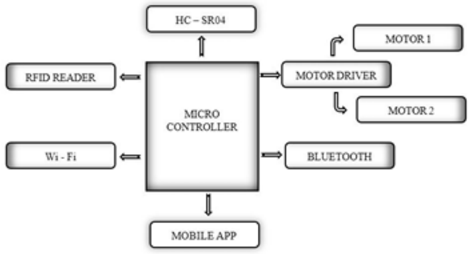
<table>
<tr><td>4.</td><td>IoT Based Smart Billing and Direction controlled Trolley [26]</td><td>

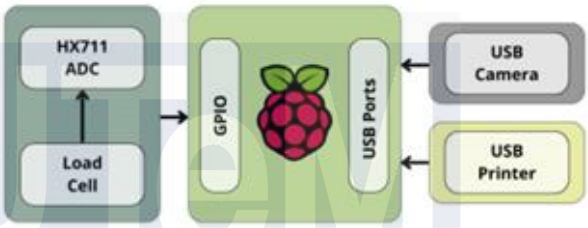
Figure 2. 9 Block diagram of IoT based Smart Billing Trolley [26]</td></tr>
<tr><td>5.</td><td>Deep Weight: A Semi-Autonomous Food Weight Billing System using Deep Learning [23]</td><td>

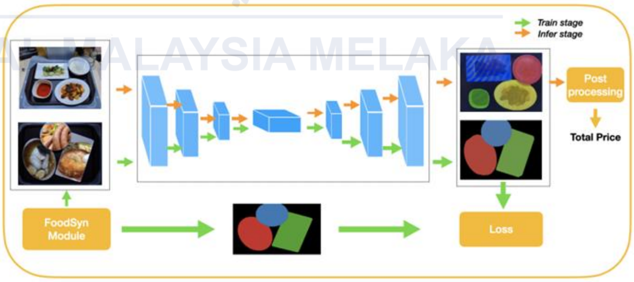
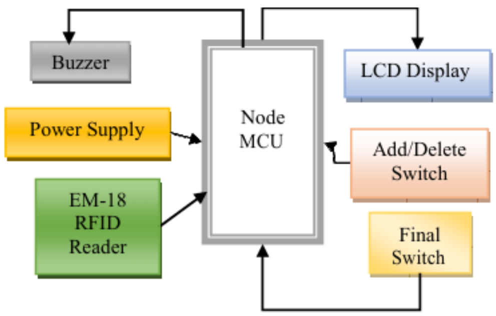
Figure 2. 10 System design of Autonomous Food Weight Billing System [23]</td></tr>
<tr><td>6.</td><td>A Real-Time Chinese Food Auto Billing System Based on Instance Segmentation [19]</td><td>

Figure 2. 11 Flowchart of Instance segmentation [19]</td></tr>
<tr><td>7.</td><td>RFID-Powered Intelligent Shopping Cart [8]</td><td>
</td></tr>
</table>

<table>
<tr><td></td><td></td><td>Figure 2. 12 Block diagram of RFID Intelligent Shopping Cart [8]

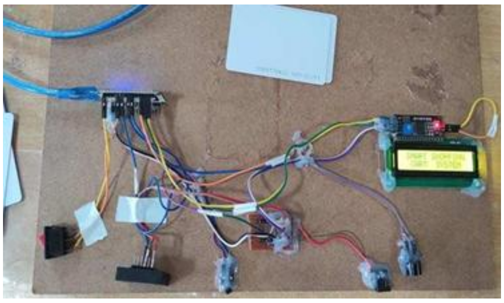



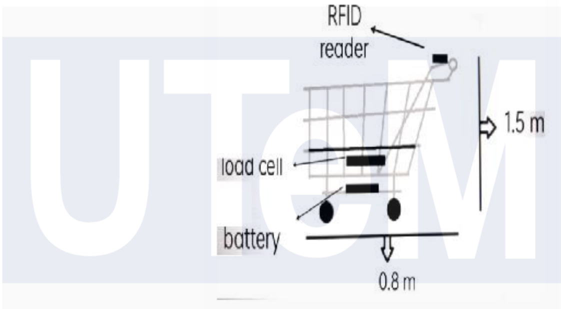
Figure 2. 13 Hardware of RFID Intelligent Shopping Cart [8]</td></tr>
<tr><td>8.</td><td>Low-Cost Automatic Purchase Cart [27]</td><td>

Figure 2. 14 Schematic of the purchase cart [27]

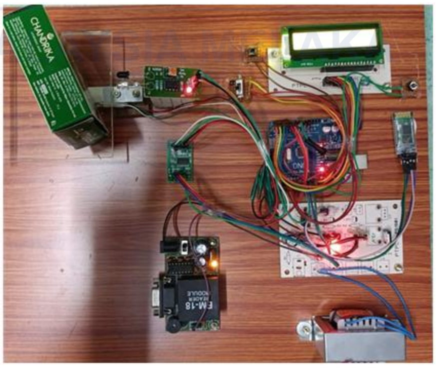



Figure 2. 15 Project of Low-Cost Purchase Cart [27]</td></tr>
</table>

2.9 Factors Related to the Work

This section focuses on identifying and analyzing the factors that influence the successful execution of the project. These factors include technical and operational that impact the development and implementation of the system.

2.9.1 Barcode Scanning Technique

A barcode scanner is a device common in smart shopping carts that is used to capture and read information encoded in barcodes. Customers can quickly scan the barcode on products on products with a barcode scanner [18]. It consists of a light source, a lens, and a light sensor translating optical impulse into electrical. The barcode readers contain decoder circuitry analyzing the barcode's image data provided by the sensor and sending the barcode's content to the scanner's output.

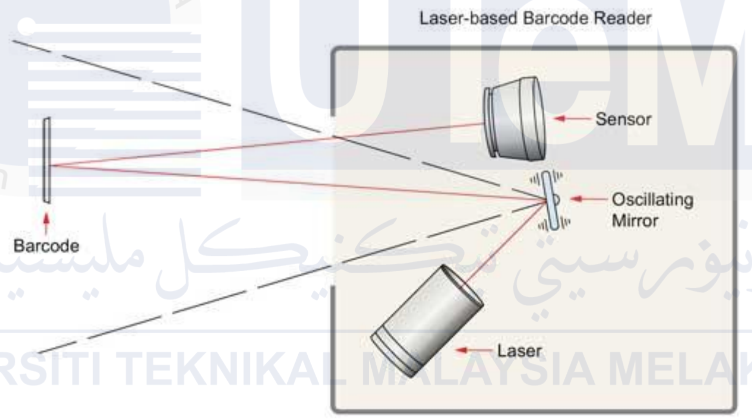


Figure 2. 16 Principle of Barcode Scanning Technique

Figure 2.16 shows a barcode scanner that uses a light source and typically employs an oscillating mirror to scan the light source back and forth across the barcode. Then, it measures the reflected light from the barcode. An analog signal is created, and it is converted into a digital signal.

There are two types of barcodes: 1D barcode (Figure 2.17) and 2D barcode (Figure 2.17). The 1D barcode is a series of black and white bars. Each bar has a different width. 2D barcodes are images of white and black dots of varying sizes. Many dots are grouped in the 2D barcode image.

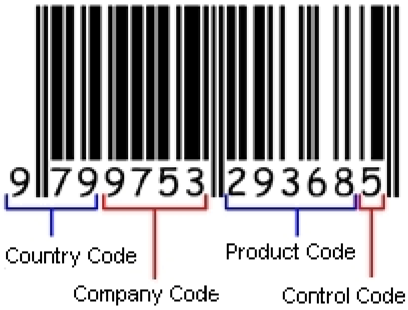


Figure 2. 17 1D Barcode

Figure 2. 18 2D Barcode

There are various standards of 1D barcodes such as UPC (EAN), Code39, and Code128 [28]. This project used 1D barcode EAN-13 for the experiment. EAN-13 is a one-dimensional barcode with a margin, a barcode symbol, a barcode, a start and stop character, data, and a check digit. This barcode contains a 12-digit UPC number, with the first three digits being the country code, the next three digits being the company code, the next five digits being the item number, and the last digits being the barcode's checking number.

2.9.2 Image Processing Technique

Image processing is a modern technology that converts image information into digital data that a computer can identify and interpret more effectively. Image processing can be roughly divided into a series of processes: image classification, compression, enhancement, coding, and feature extraction [29]. To ensure that the image is clear and properly recognizes information, image processing should be supported by various technologies to improve the resolution and quality of the image. The development of artificial intelligence technology also promotes the development of image processing technology, which makes image processing technology widely used in areas such as pattern recognition, machine vision, and multimedia technology.

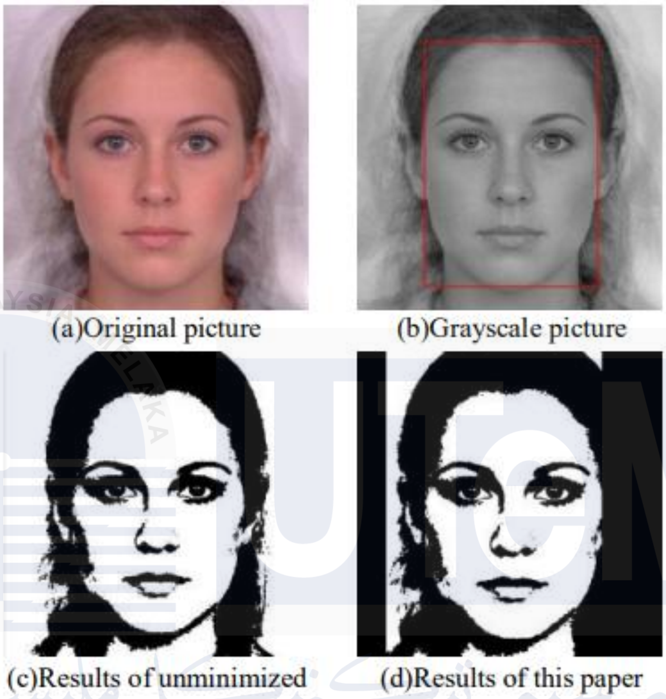


Figure 2. 19 The process of Image Processing [30]

2.9.3 Weighing Technique

The weighing technique refers to the method or procedure used to determine the weight of an object. It involves the use of tools or equipment, such as scales or load cells, and specific processes to ensure accurate and reliable weight measurements. Common techniques include static weighing, where the weight of an object is measured while stationary, dynamic weighing, which involves measuring moving objects, is often used in conveyor belt systems.

Figure 2. 20 Smart restaurant using conveyor belt [31]

The tare and gross weight method is another widely used approach, where the weight of a container is subtracted from the total weight to obtain the net weight of the product. In the automated self-checkout system, the weighing technique involves calibrating the load cell sensor with known reference weights to ensure accuracy, applying tare adjustment to exclude the weight of the platform or container, and employing static weighing to measure individual product weights.

2.10 Summary

Therefore, analyzing the literature, it is crucial to address the problem of using automatic self-billing counters in hypermarkets and supermarkets. Traditional counters are inconvenient and time-consuming, require employees to interact directly with customers, involve a certain level of risk, and do not meet modern demands. IoT technologies like RFID, deep learning, and image processing are installed in self-billing counters to solve the problem since they require fewer human resources and less time to reduce mistakes. Other research has shown that such systems can enhance organizational effectiveness and productivity as well as its customers. Instead of just reinforcing the act of checking out, these technologies complement inventory tracking and protection, making the self-billing counter an excellent innovation in the retail industry.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter outlines, in a step-by-step approach, how the automated self-billing counter system was applied with the Raspberry Pi combined with a barcode scanner and image processing technologies. This chapter explains the development and advancement of the systems present.

3.2 Development of an Automated Self-Billing Counter with a Barcode Scanner and Image Processing Using Raspberry Pi

The automated self-billing counter using Raspberry Pi incorporates a barcode scanner and image processing to streamline the final purchase process step. Using Raspberry Pi as the primary hardware platform, the system efficiently scans product information through barcode scanning. Additionally, a Pi Camera is employed to detect the product before measuring its weight. Receipts are sent to customers via WhatsApp, contributing to environmental sustainability by reducing paper waste. This innovative solution enhances the shopping experience by minimizing the time required for billing compared to manual methods, which are more prone to human error.

3.3 Block diagram of Automated Self-Billing Counter

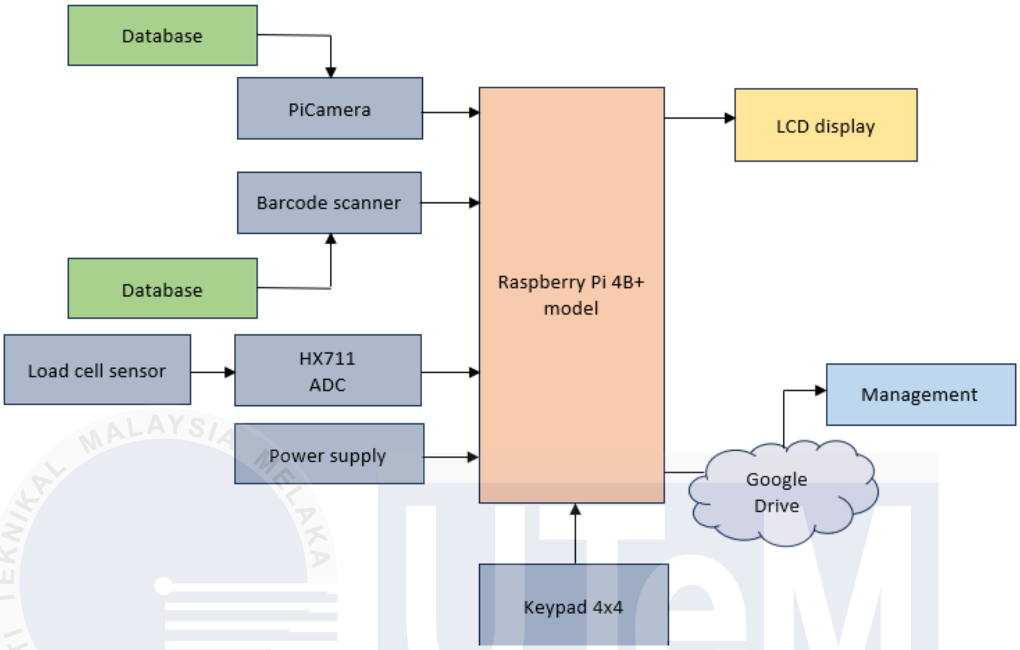


Figure 3. 1 Block diagram of An Automated Self-Billing Counter

Figure 3.1 shows the block diagram of the automated self-billing counter system. This system includes a Pi Camera, barcode scanner, load cell sensor, HX711 ADC module, and Google Drive as a cloud system. The most essential component is the Raspberry Pi, which is set up as a root node of the system. It uses a GPIO (General Purpose Input/Output) port to connect it to the Raspberry Pi. Hardware inputs include a Raspberry Pi Camera for capturing images, a barcode scanner for reading barcode data, and a load cell sensor connected to HX711 ADC for weight measurements. The power supply is necessary for constant draws of the Raspberry Pi and such devices as it controls. Program data is stored in a database in Google Drive that controls the data flows and system processes. Real-time outputs include the LCD that displays the message inputted by the user. This integrated setup is ideal for applications in inventory control or other retailing environments because the system's ability to identify the objects and weights of objects is more accurate.

3.4 Flowchart of Automated Self-Billing Counter

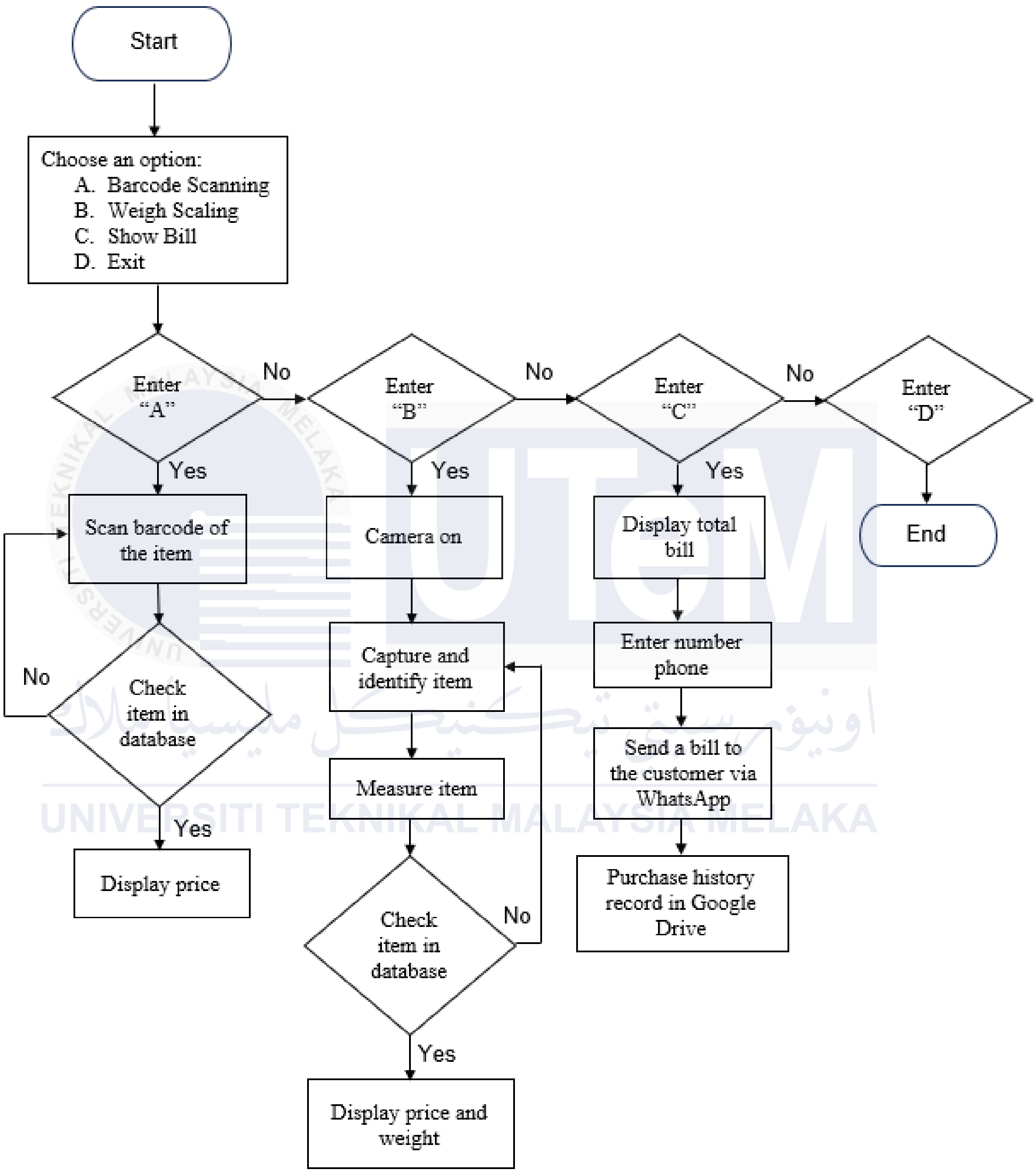


Figure 3. 2 Flowchart of An Automated Self-Billing

Figure 3.2 shows the automated self-billing counter system begins with a user selecting an option from the main menu, which includes A. Barcode Scanning, B. Weigh Scaling, C. Show bill, or D. Exit. Upon choosing option A, the system scans the barcode of the product. It then checks if the product exists in the database. If the product is found, the price is displayed.

Otherwise, the system loops back to allow the user to scan another barcode.

If option B is selected, the system activates the camera to capture and identify the product. The product is then measured to determine its weight. The database is checked for the product's details, and if found, the price and weight are displayed. If not, the process loops back to capture the product again.

Selecting option C, the system calculates and displays the total bill. The user to enter their phone number. A digital copy of the bill is sent to the customer via WhatsApp, and the purchase history is recorded in Google Drive for future reference. Finally, choosing option D terminates the process. This systematic flow ensures efficiency, accuracy, and an eco-friendly approach by reducing paper usage through digital receipts.

3.5 Experimental Setup

The structure of this work is designed to give a new and integrated analytical solution to develop the system more accurately and reliably. This system employs several technologies, including hardware and software.

3.5.1 Hardware

This section explained the function of hardware in the automated self-billing system.

3.5.1.1 Raspberry Pi 4B+ model

The Raspberry Pi 4B+ model is a powerful credit card-sized minicomputer, that serves as the central processing unit and primary hardware platform for this project. Renowned for its versatility and efficiency, the Raspberry Pi 4B+ is capable of interfacing seamlessly with various input and output devices, making it an ideal choice for implementing an automated system. In this project, the Raspberry Pi is connected to an HX711 load cell sensor, which facilitates accurate weight measurements, and a Pi Camera, which is used for capturing images and identifying products through image processing. Additionally, an LCD display is integrated into the system to provide real-time feedback to users, such as displaying product details, weight, and prices. Together, these components enable the Raspberry Pi to act as the “brain: of the project, coordinating all operations and ensuring smooth functionality. This combination of hardware and software highlights the Raspberry Pi’s role in creating an efficient and reliable self-billing system.

Figure 3. 3 Raspberry Pi 4B+

3.5.1.2 Raspberry Pi Camera (Pi Camera)

The Raspberry Pi Camera (Figure 3.4) is a high-performance camera module designed to integrate seamlessly with Raspberry Pi boards, enabling the capture of high-quality images and video. In this project, the Pi Camera plays a crucial role in the **weight** scaling process. When the user selects the “weigh scaling”, the Pi Camera is activated to capture and identify the product placed on the plate of the HX711 load cell sensor.

Figure 3. 4 Pi Camera

The camera captures an image of the product, which is then processed to detect and identify the product. This identification step ensures that the system can match the product with its corresponding details, such as weight and price that are stored in the database. Once the product is successfully identified, the system proceeds to measure its weight using a load cell sensor and HX711 module. This combination of image capture and weight measurement ensures accurate detection and processing of the products.

3.5.1.3 HX711 Load Cell Sensor

This system incorporates an HX711 load cell sensor to accurately measure the weight of products. The straight bar load is a key component, as it translates the applied force or pressure into an electrical signal. This design allows for precise force measurements, making it suitable for use in weighing applications. Since the electrical signal output by the load cell is very small, the HX711 acts as an amplifier to enhance the signal and enable accurate readings. In this project, the load cell sensor is specifically utilized to weigh items within a weight range of 1 kg to 5 kg. The setup of the load cell sensor and HX711 including their configuration is detailed in Table 3.1, while their connection to Raspberry Pi is illustrated in Table 3.2. This integration ensures that the system can reliably measure and process weight data, contributing to the overall functionality of the automated self-billing counter.

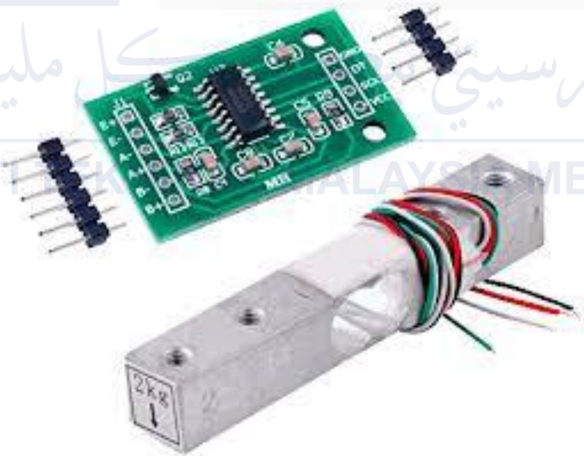


Figure 3. 5 Load cell sensor

Table 3. 1 Load cell sensor to HX711

Load cell	HX711
E+	Red wire
E-	Black wire
A-	Green wire
A+	White wire

Table 3. 2 HX711 to Raspberry Pi

HX711	Raspberry Pi
GND	GND
DT	Pin 26
SCK	Pin 16
VCC	5V

The load cell sensor and HX711 module play a critical role in the weighing scaling process. When the user selects the weighing scaling, the camera is activated to capture and identify the product. Once identified, the product is placed on the load cell sensor to measure its weight. The load cell sensor is equipped with strain gauges, detects the applied force from the product's weight, and generates a small electrical signal proportional to the force. The HX711 module is used to amplify the electrical signal and convert it into a digital. The module ensures that the weight measurement is both accurate and precise.

The Raspberry Pi reads the digital output from the HX711 and checks the database to match the product's weight and details. The Raspberry Pi measures its weight and calculates the total price by multiplying the weight by the predefined price per gram. If the product is found in the database, the system calculates and displays the weight and corresponding price on the LCD screen. Otherwise, if the product is not found, the system loops back and allows the user to repeat the process.

3.5.1.4 Barcode Scanner

A barcode scanner is an essential electronic device used in this project to read and interpret barcodes printed on product labels. Barcodes are machine-readable representations of information, typically displayed as a series of black and white lines or intricate patterns such as 2D barcodes. In this system, the barcode scanner is responsible for decoding the information embedded in the barcode and transmitting it to the Raspberry Pi for further processing.

Figure 3. 6 Barcode scanner

When the user selects the barcode scanning option from the main menu, the scanner is activated to read the barcode on the product. The system then checks the scanned information against its database to retrieve details such as the product's name and price. If the product exists in the database, the corresponding details are displayed on the LCD display. If the product is not found, the system prompts the user to scan again. This barcode scanning functionality ensures efficient and accurate retrieval of product information and enhances the overall user experience.

3.5.1.5 Keypad 4x4

The 4x4 keypad in this project serves as the primary input interface, allowing users to interact with the self-billing counter. This compact and versatile keypad consists of 16 keys arranged in a 4-row by the keypad to navigate through the menu options including selecting barcode scanning, weigh scaling, show bills, or exit.

Figure 3. 7 4x4 keypad

During the barcode scanning process, the keypad allows users to input specific commands such as scanning another product. The weight scaling process enables the user to proceed with product detection and weight measurement. Additionally, the keypad facilitates the input of the phone number during the show bill process, ensuring that the digital receipt is sent to the correct customer via WhatsApp. The 4x4 keypad's integration with the Raspberry Pi provides a simple and reliable way for users to operate the system.

3.5.1.6 LCD display

The LCD used in the system is a 20x4 character display, which allows for a clear and concise presentation of information to the user. As the primary output interface, the LCD displays essential details during the operation of the self-billing counter system. When the system detects or scans a product on the counter, the LCD shows real-time information about the product including the name of the product, its price, and weight.

Figure 3. 8 LCD display

In the barcode scanning process, the LCD displays the product's name and price once the barcode is successfully scanned and matched with the database. Similarly, during the weight scaling process, the LCD presents the product's weight and price after the system identifies the product using a camera and measures its weight with the load cell sensor. The use of the 20x4 LCD ensures that all relevant information is easily readable by the user, contributing to the system's user-friendly design and enhancing the efficiency of the automated self-billing counter.

3.5.2 Software

This section explained the function of software used in the automated self-billing system.

3.5.2.1 Raspberry Pi Software

In this project, the Raspberry Pi is programmed using the Raspberry Pi software, which facilitates the development and execution of code on the device. This software serves as an integral component of the integrated development environment (IDE) typically used on a PC to create and manage applications for the Raspberry Pi. The programming language chosen for this project is Python, renowned for its simplicity, versatility, and extensive library support, making it ideal for embedded systems and hardware interfacing.

Python is utilized to implement the various functionalities of the system, including interfacing with the barcode scanner, HX711 load cell sensor, Pi Camera, and LCD as well as managing the 4x4 keypad for user input. The code enables the seamless integration of these components, allowing the Raspberry Pi to perform tasks such as scanning barcodes, detecting and weighing products, processing product data, and displaying outputs on the LCD. This programming approach ensures the efficient operation of the automated self-billing counter system.

Figure 3. 9 Raspberry Pi Imager

3.5.2.2 Teachable machine software

Teachable Machine is an online tool developed by Google that allows users to train machine learning models without needing extensive programming knowledge. It provides a simple and intuitive interface where users can upload images, sound clips, or pose data to teach the model how to recognize specific objects, actions, or patterns. Once trained, the models can be deployed directly in browsers, making them accessible for applications like smart devices, mobile apps, or web-based projects. Teachable Machine supports various types of machine-learning models, including image classification, audio recognition, and pose estimation.

Figure 3. 10 Teachable Machine

Teachable Machine utilized TensorFlow as the fundamental foundation for building and training machine learning models. TensorFlow is a Google-developed open-source machine-learning library that includes algorithms and tools for creating and optimizing machine-learning models.

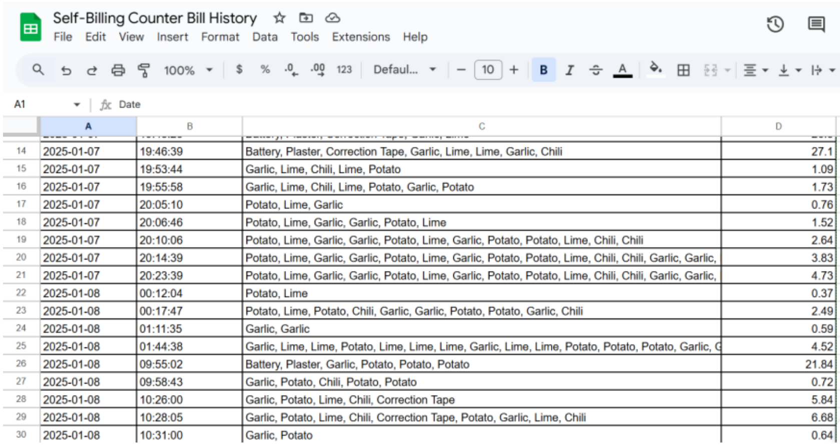
As part of the system’s image processing, the models used are MobileNet and Convolution Neural Network (CNN). The four products were selected for both training and testing: garlic, lime, potato, and chili. These products serve as the primary focus for developing the machine-learning model, allowing it to learn and recognize features from each product. By using these specific objects, the model is trained to accurately classify and identify them based on visual cues. The training process ensures that the model gains the necessary insight to differentiate between the four products, while the testing phase evaluates its performance in real-world scenarios.

3.5.2.3 Google Drive

Google Drive is a cloud-based storage service developed by Google that allows users to store, share, and access files securely from any device with an internet connection. It offers both free and paid storage options, where users can upload various types of files, including documents images, videos, and more. Google Drive provides features like real-time collaboration, making it easy for multiple users to work on the same file simultaneously, with access control and version history for better organization and tracking.

Figure 3. 11 Google Drive

In this project, the system emailed the customer a digital receipt for the transaction via WhatsApp. The purchase history is recorded in Google Drive for administration tracking. The Google spreadsheets have four elements which are the date, time, the item purchased, and the amount of bill.



	A	B	C	D
14	2025-01-07	19:46:39	Battery, Plaster, Correction Tape, Garlic, Lime, Lime, Garlic, Chili	27.1
15	2025-01-07	19:53:44	Garlic, Lime, Chili, Lime, Potato	1.09
16	2025-01-07	19:55:58	Garlic, Lime, Chili, Lime, Potato, Garlic, Potato	1.73
17	2025-01-07	20:05:10	Potato, Lime, Garlic	0.76
18	2025-01-07	20:06:46	Potato, Lime, Garlic, Garlic, Potato, Lime	1.52
19	2025-01-07	20:10:06	Potato, Lime, Garlic, Garlic, Potato, Lime, Garlic, Potato, Potato, Lime, Chili, Chili	2.64
20	2025-01-07	20:14:39	Potato, Lime, Garlic, Garlic, Potato, Lime, Garlic, Potato, Potato, Lime, Chili, Chili, Garlic, Garlic,	3.83
21	2025-01-07	20:23:39	Potato, Lime, Garlic, Garlic, Potato, Lime, Garlic, Potato, Potato, Lime, Chili, Chili, Garlic, Garlic,	4.73
22	2025-01-08	00:12:04	Potato, Lime	0.37
23	2025-01-08	00:17:47	Potato, Lime, Potato, Chili, Garlic, Garlic, Potato, Potato, Garlic, Chili	2.49
24	2025-01-08	01:11:35	Garlic, Garlic	0.59
25	2025-01-08	01:44:38	Garlic, Lime, Lime, Potato, Lime, Lime, Lime, Garlic, Lime, Lime, Potato, Potato, Potato, Garlic, G	4.52
26	2025-01-08	09:55:02	Battery, Plaster, Garlic, Potato, Potato, Potato	21.84
27	2025-01-08	09:58:43	Garlic, Potato, Chili, Potato, Potato	0.72
28	2025-01-08	10:26:00	Garlic, Potato, Lime, Chili, Correction Tape	5.84
29	2025-01-08	10:28:05	Garlic, Potato, Lime, Chili, Correction Tape, Potato, Garlic, Lime, Chili	6.68
30	2025-01-08	10:31:00	Garlic, Potato	0.64

Figure 3. 12 The transaction history in Google Spreadsheet

3.6 An Automated Self-Billing Counter with Barcode Scanning and Image Processing Prototype

An automated self-billing counter was developed to replicate the functionality of a self-service billing system, integrating crucial hardware and software components for efficient operation. It features a compact and cost-effective physical structure, with a barcode scanner located at the left for effective product scanning. An LCD display is mounted on the front panel to present real-time information on scanned products, including product details, prices, and the total bill amount. A keypad is provided for users to input selections, confirm transactions, or navigate system options. A load cell sensor is placed under the plate to measure the weight of products which can be used to verify the accuracy of the products or calculate prices of products by weight. The central processing unit is a Raspberry Pi that incorporates barcode scanning, weighing process, and image processing functionalities to automate the billing process.

Figure 3. 13 Prototype of an automated self-billing counter

3.7 Summary

Finally, this section articulates the design of an Automated Self-Billing Counter System through Raspberry Pi technology combined with Barcode scanning and Image processing that will help increase the billing system speed to make it fully automated in retail stores. To make the system, the Raspberry Pi is used as the central processing unit, with the assistance of the Pi Camera for image processing to find products that do not have barcodes, the barcode scanner to enter the information about a product, and the load cell sensor connected to the HX711 ADC for accurate weight measurement. A flowchart illustrates the operation procedure of the system in which fruits/vegetables are separated and sorted according to their respective barcodes to prevent overlapping and to facilitate proper assessment of their weight and prices. The experimental configuration calls for using additional pieces of hardware such as a load cell sensor, a Raspberry Pi camera, an LCD screen, and software components, including pre-installed software for Raspberry Pi and Google Drive for data analysis, automation of the process, and user interface to improve output rates, waiting time, and control over possible human incompetence.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter highlights the deployment of the Raspberry Pi-based system is comprehensively examined, encompassing hardware configuration, software integration, and real-world testing. The core components, namely the Raspberry Pi 4B+, Pi Camera, a barcode scanner, and a load cell sensor are strategically employed. Python programming plays a crucial role in implementing functionalities such as barcode scanning, weigh scaling process, and the assessment of system accuracy through machine learning. Real-world testing involves nine products, offering a robust evaluation of the system's reliability, accuracy, and user-friendly across three scenarios. The ensuing results, including accuracy metrics derived from confusion matrix analysis, furnish valuable insight into the practical implications of the developed system.

4.2 Results and Analysis

This project entitled "Development of An Automated Self-Billing Counter with a Barcode Scanning and Image Processing on Raspberry Pi" is completed and the results are satisfactory. With the completion of the project have received good results. This work was inspired by a demonstration. It may be easily adopted by those who wish to make adjustments or add new features. Overall efficiency will increase, the number of people needed to staff markets will decrease, and customer wait times will be cut or reduced thanks to this effort.

4.2.1 Preliminary Result

In the preliminary results of an automated self-billing counter simulations were conducted using Proteus software to evaluate the system functionally. Figure 4.1 demonstrates the use of Proteus's Visual Design to test the image processing simulation. In this step, the Pi Camera module is employed to capture images of products placed on the plate, while a Raspberry Pi serves as the central processing unit for the system. The result in Figure 4.1 illustrates the outcome of image processing, where the system successfully identifies and recognizes a red apple as the target object.

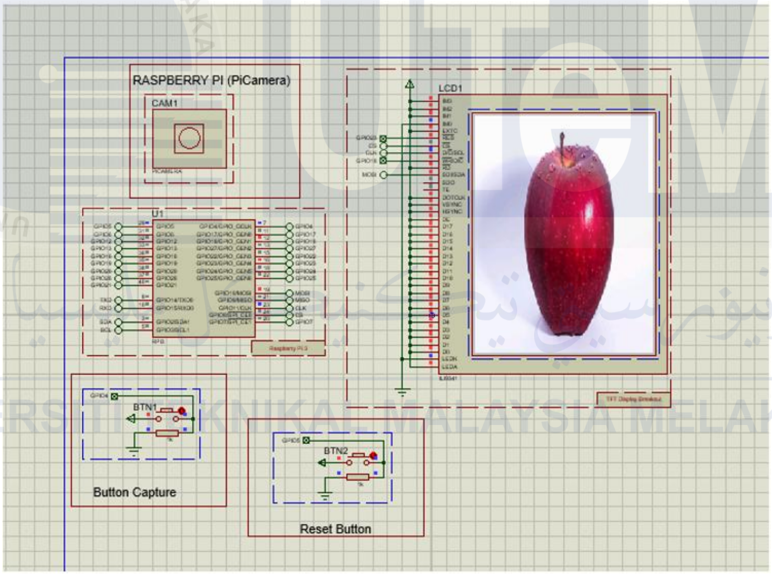


Figure 4. 1 Preliminary result of image processing

The system is equipped with two primary buttons, labeled "Button Capture" and "Reset Button" to facilitate its operation. When the button capture is pressed, the Pi Camera activates to capture an image of the product and display it on the connected LCD displays for further processing. On the other hand, pressing the reset button clears the display and resets the system, allowing it to repeat the process and capture a new image of the next product placed on the plate.

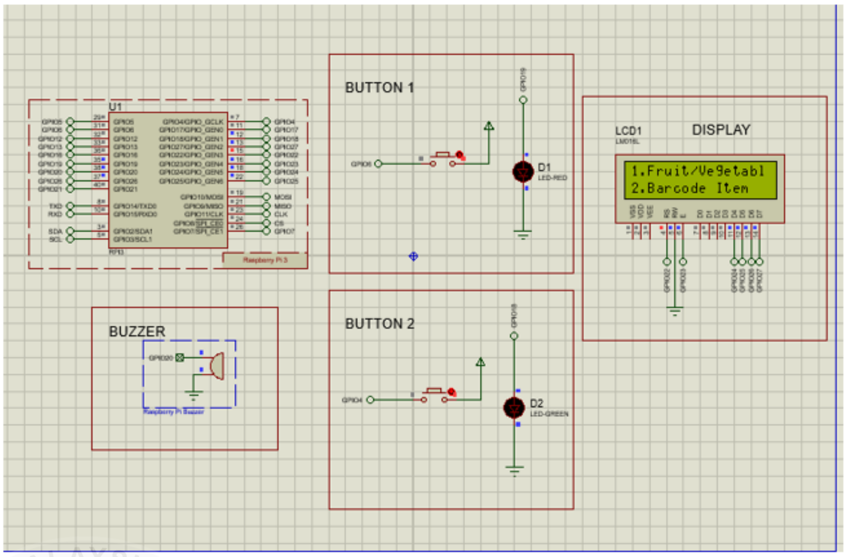


Figure 4. 2 Preliminary result of the project

Figure 4.2 provides a detailed simulation of the automated self-billing counter incorporating various components, including a Raspberry Pi, LCD display, two push buttons, and a buzzer. These push buttons labeled as Button 1 and Button 2 serve as input controls. Specifically, button 1 initiates the barcode scanning process. Once button 1 is pressed, the system displays the product's name and price on the LCD. The button 2 activates the image processing mode. When button 2 is pressed, the Pi Camera captures an image of the product. Simultaneously, an LED light turns on to indicate the camera operation, and a buzzer sounds to confirm the action. Following the successful detection and recognition of the product by the Pi Camera, the system calculates and displays the product's weight, name, and price. This simulation continues to operate seamlessly until manually terminated.

4.2.2 An Automated Self-Billing Counter with Barcode Scanning and Image Processing

In terms of performance, the prototype successfully demonstrated its capability to scan barcodes, measure product weights using a load cell sensor, identify the product using image processing, and process billing information efficiently. The findings indicated that the system was capable of updating the entire bill amount in **real-time** while displaying relevant details on the LCD, hence ensuring a seamless user experience. The load cell sensor added an extra layer of functionality by enabling weight-based verification, hence improving the system's reliability. During development and testing, challenges such as optimizing the response time of the barcode scanner, ensuring reliable image processing, and calibrating the load cell sensor for accurate measurements were encountered and addressed through iterative testing. Overall, the prototype meets the objectives of the project, offering an automated solution that improves the efficiency and convenience of the billing process, with room for future enhancements to make the system more robust and versatile.

Figure 4. 3 The prototype

4.2.3 Experiment Result

This section presents the results obtained from the development and testing of the automated self-billing counter in various scenarios. The system integrates barcode scanning and image processing on a Raspberry Pi platform to streamline the billing process. Experiments were conducted to evaluate the system’s performance in terms of accuracy, efficiency, and reliability. The outcomes demonstrate the effectiveness of the proposed solution in automating billing tasks and enhancing user convenience.

4.2.4 Experiment Result of Barcode Scanning

This section presents the results of the barcode scanning experiments conducted to evaluate the performance of the system. The experiment was designed to assess the reliability of the barcode scanner in identifying product information under various conditions. The findings provide insights into the effectiveness of the barcode scanning component in contributing to the overall functionality of the automated self-billing counter.

4.2.4.1 Barcode Scanning

This project was tested using a barcode reader to evaluate its performance with five distinct items, each equipped with a barcode. The selected items included a box of plasters, a correction tape, a hair tie, a battery, and a face mask. These items, as depicted in Figure 4.4, were utilized to assess the system’s capability to scan and identify barcoded items accurately.

Figure 4. 4 Five products with barcode labels

The barcode scanning functionality implemented in this project has been tested using a set of five items, each equipped with unique barcodes. For each item, data acquisition and experimentation were conducted 50 times as shown in Figure 4.5 to thoroughly evaluate the reliability and accuracy of the system in correctly identifying and processing the barcodes. The results of these tests are presented in Figure 4.6 demonstrating consistent performance, with each item achieving close to 100% reliability, indicating the high accuracy and dependability of the barcode scanning system. This consistency underscores the robustness of the system in identifying and processing barcodes across various item types.

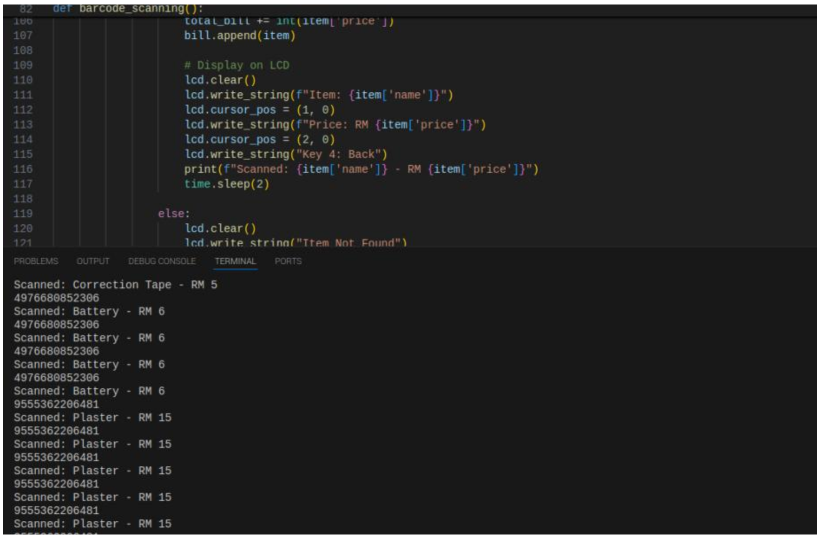


Figure 4. 5 The output of barcode scanning

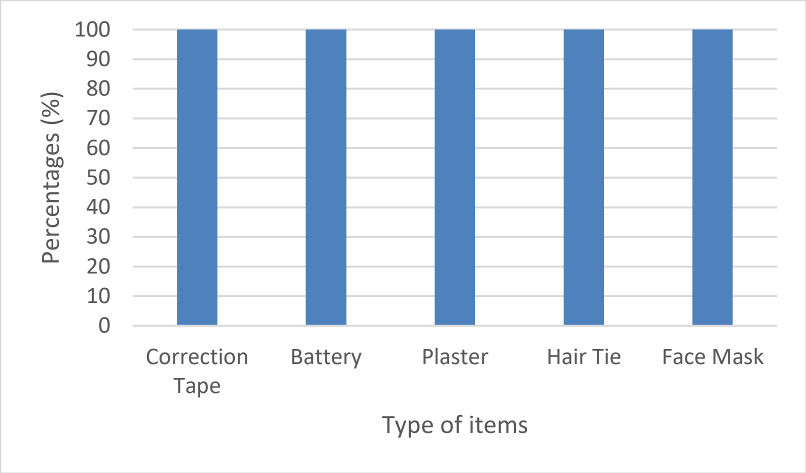


Figure 4. 6 The reliability of barcode scanning

Figure 4.7 illustrates the output generated by the barcode scanner, which is displayed on an LCD screen integrated into the system. Upon successfully scanning the barcode of an item, the Raspberry Pi processes the data and outputs the corresponding details, including the name of the item and its total price. This demonstrates the system's capability to accurately decode and display relevant information in real-time, ensuring a seamless user experience.

Figure 4. 7 The output of barcode scanning in LCD

4.2.5 Experiment Result of Image Processing

This section discusses the results of the image processing experiments conducted to evaluate the system's ability to recognize and verify products. The experiments focused on assessing the accuracy, and the precision of the image processing component under different lighting conditions, angles, and product variations. The outcomes highlight the effectiveness of the image processing feature in enhancing the functionality and accuracy of the automated self-billing counter.

4.2.5.1 Training the model

The machine learning model was trained using a dataset created with Teachable Machine, consisting of a total of 120 images. The dataset was evenly distributed, containing 30 images each of chili, garlic, lime, and potato. The images were captured using the Pi Camera, with the items placed at various angles and distances to enhance the robustness of the model. Significant lighting issues were encountered during the initial data collection process, which impacted the quality of the dataset. To address this challenge, the dataset was recollected under varying conditions of brightness and exposure, as illustrated in Figure 4.8. This ensured a diverse and balanced dataset, enabling the model to perform reliably across different environmental conditions.

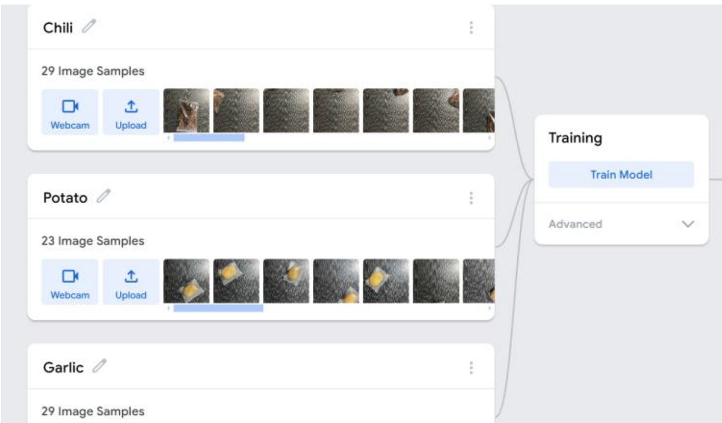


Figure 4. 8 Train data in Teachable Machine

4.2.5.2 Testing the model

The result in Figure 4.9 indicates that the model successfully classified the tested item with a high degree of accuracy. Any misclassifications observed were primarily attributed to extreme variations in lighting or significant occlusion of the item during testing. These findings demonstrate the effectiveness of the model in image classification while also highlighting areas for potential improvement such as increasing dataset diversity or incorporating preprocessing techniques to mitigate lighting effects.

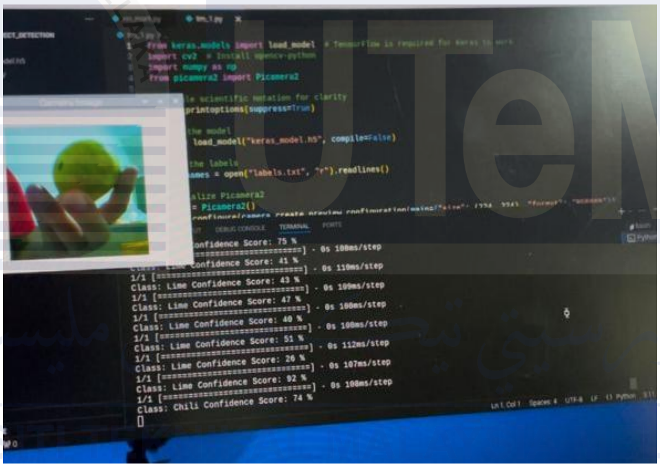


Figure 4. 9 Testing lime to detect in image processing

4.2.5.3 Explanation of Confusion Matrix

A confusion matrix, often referred to as an error matrix, is a matrix that provides information about the classification results of a Machine Learning model to compare to the actual outcomes. Currently, the system is designed to classify four distinct items: potato, lime, chili, and garlic. There are four elements in the confusion matrix:

a) **True Positive (TP):** The model correctly predicts the positive class.

b) **True Negative (TN):** The model correctly predicts the negative class.

c) **False Positive (FP):** The model incorrectly predicts the positive class when the actual class is negative.

d) **False Negative (FN):** The model incorrectly predicts the negative class when the actual is positive.

These categories help in calculating various performance metrics such as accuracy, precision, recall, and specificity, which provide insights into the effectiveness of the classification model.

4.2.5.4 Potato Confusion Matrix

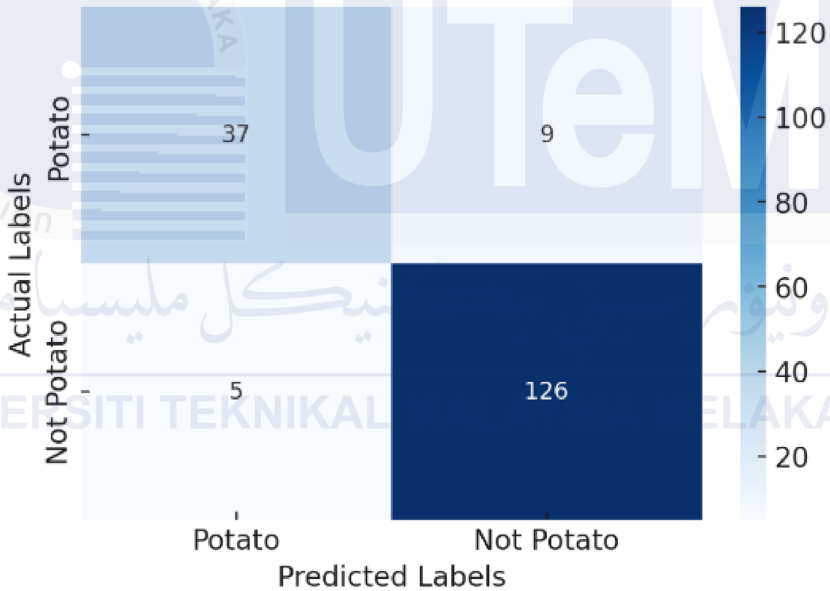


Figure 4. 10 Potato confusion matrix

Table 4. 1 Potato confusion matrix calculation

Information	Calculation
• True Positive (TP): 37 • True Negative (TN): 126 • False Positive (FP): 5 • False Negative (FN): 9	$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$ $Accuracy = \frac{(37 + 126)}{(37 + 126 + 5 + 9)}$ $Accuracy = 0.9209 \times 100\ \%$ $Accuracy = 92.09\%$ $Precision = \frac{TP}{(TP + FP)}$ $Precision = \frac{37}{(37 + 5)}$ $Precision = 0.8809 \times 100\%$ $Precision = 88.09\%$

4.2.5.5 Garlic Confusion Matrix

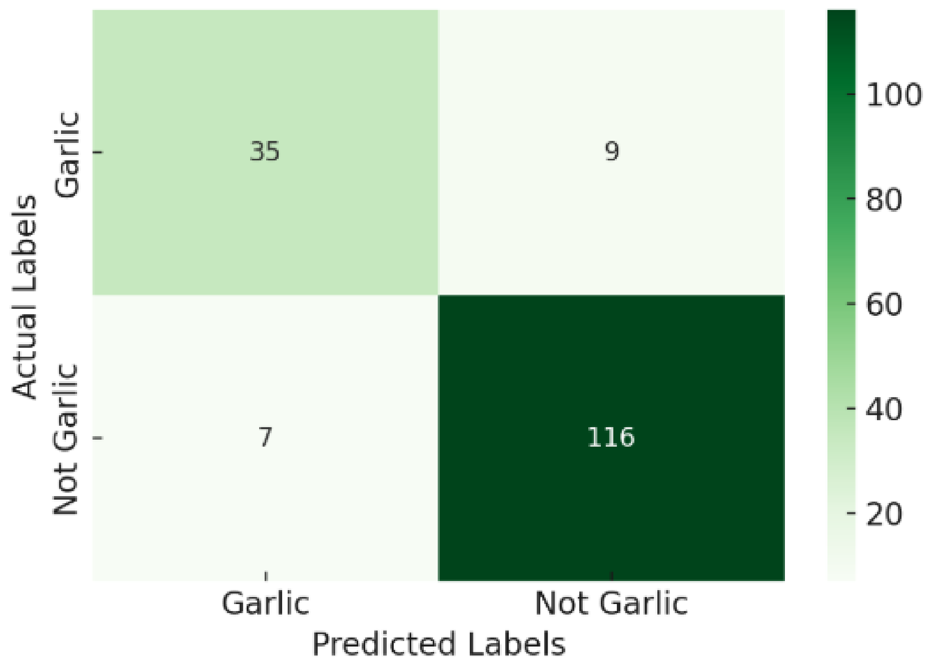


Figure 4. 11 Garlic confusion matrix

Table 4. 2 Garlic confusion matrix calculation

Information	Calculation
• True Positive (TP): 35 • True Negative (TN): 116 • False Positive (FP): 7 • False Negative (FN): 9	$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$ $Accuracy = \frac{(35 + 116)}{(35 + 116 + 7 + 9)}$ $Accuracy = 0.9042 \times 100\%$ $Accuracy = 90.42\%$ $Precision = \frac{TP}{(TP + FP)}$ $Precision = \frac{35}{(35 + 7)}$ $Precision = 0.833 \times 100\%$ $Precision = 83.30\%$

4.2.5.6 Lime Confusion Matrix

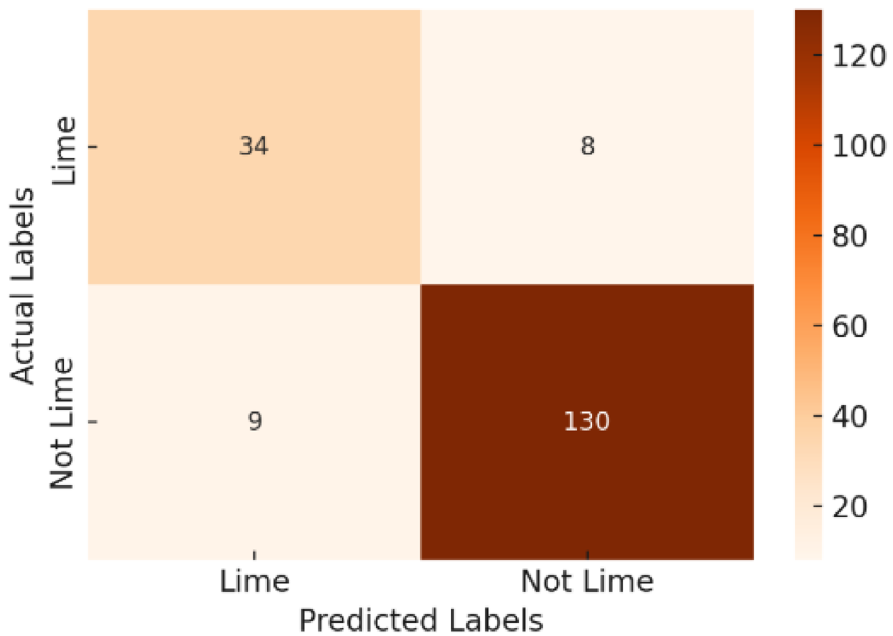


Figure 4. 12 Lime confusion matrix

Table 4. 3 Lime confusion matrix calculation

Information	Calculation
• True Positive (TP): 34 • True Negative (TN): 130 • False Positive (FP): 9 • False Negative (FN): 8	$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$ $Accuracy = \frac{(34 + 130)}{(34 + 130 + 9 + 8)}$ $Accuracy = 0.9061 \times 100\%$ $Accuracy = 90.61\%$ $Precision = \frac{TP}{(TP + FP)}$ $Precision = \frac{34}{(34 + 9)}$ $Precision = 0.7907 \times 100\%$ $Precision = 79.07\%$

4.2.5.7 Chili Confusion Matrix

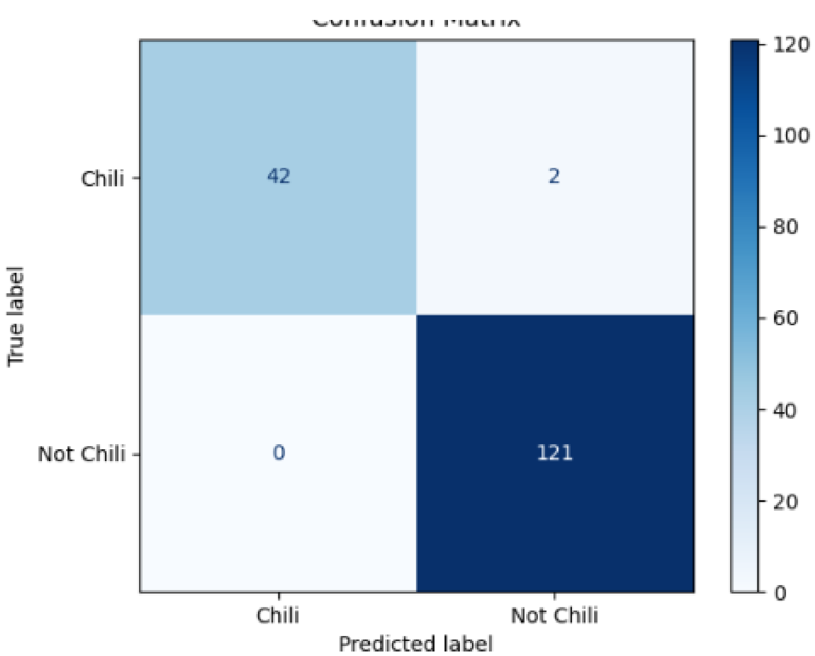


Figure 4. 13 Chili confusion matrix

Table 4. 4 Chili confusion matrix calculation

Information	Calculation
• True Positive (TP): 35 • True Negative (TN): 121 • False Positive (FP): 7 • False Negative (FN): 9	$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$ $Accuracy = \frac{(35 + 121)}{(35 + 121 + 7 + 9)}$ $Accuracy = 0.9070 \times 100\%$ $Accuracy = 90.70\%$ $Precision = \frac{TP}{(TP + FP)}$ $Precision = \frac{35}{(35 + 7)}$ $Precision = 0.833 \times 100\%$ $Precision = 83.30\%$

4.2.6 Experiment Result of The Weighing Process

The weighing process experiment was conducted to evaluate the accuracy and reliability of the weight measurement system in the automated self-billing counter in two scenarios. This system utilizes a load cell sensor connected to an HX711 amplifier, integrated with a Raspberry Pi 4B+ to capture and process weight data. The primary objective of this experiment is to assess the system’s performance under different scenarios, including variations in product weight and environmental conditions. The results provide insights into the system’s precision, calibration efficiency, and an average of each product for improvement to accurate billing in real-world applications.

4.2.6.1 Scenario 1: Weighing process calibration

This project incorporates a load cell sensor in conjunction with the HX711 module to facilitate the weighing of items during the checkout process. The HX711 module is employed to manage and amplify the signals from the load cell sensor, ensuring accurate weight measurement. Before conducting any weighing operations, the load cell sensor undergoes a calibration process to determine the tare value. The tare value represents the officially recognized weight of the empty plate excluding any additional load and serves as a baseline for precise weight. The tare value of this project is -277691.0 of the empty plate as shown in Figure 4.14.

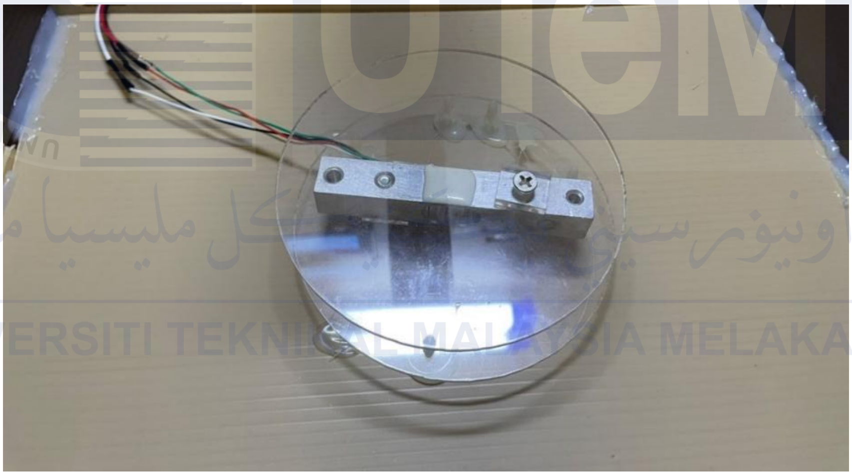


Figure 4. 14 Load cell sensor with plate

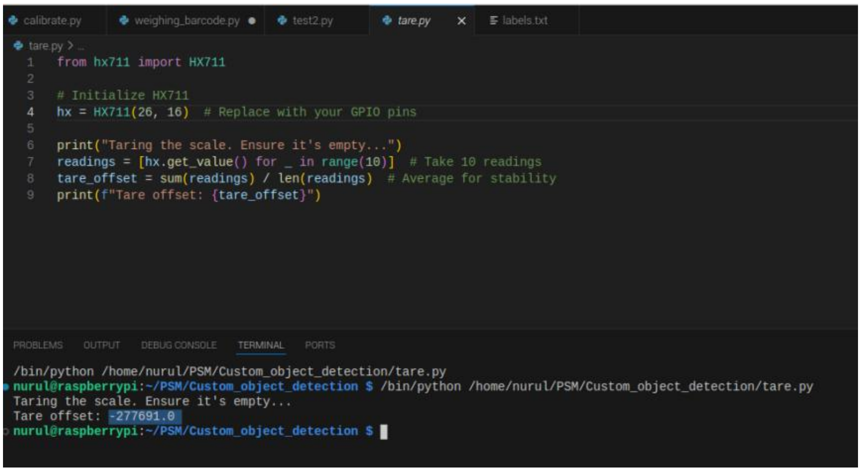


Figure 4. 15 The tare value

The reference unit plays a critical role in converting the raw digital signal generated by the load cell into a meaningful weight measurement unit. In this project, the reference unit value as shown in Figure 4.16 has been determined to be -438.9425287356322 and the known weight used is 90 grams. Both the tare value and the reference unit are essential parameters for ensuring accurate weight measurement, as they establish the baseline and scaling factors required for precise calculations during the weighing process.

$$Reference\ unit = \frac{Raw\ mean\ weight - Tare\ value}{Known\ weight}$$

Figure 4. 16 Reference value

4.2.6.2 Scenario 2: Weighing process on the product

Figure 4.17 shows the weighing scale was tested using four different items: garlic, potato, lime, and chili. For each item, the measured weight value was recorded 50 times to ensure accuracy and reliability.

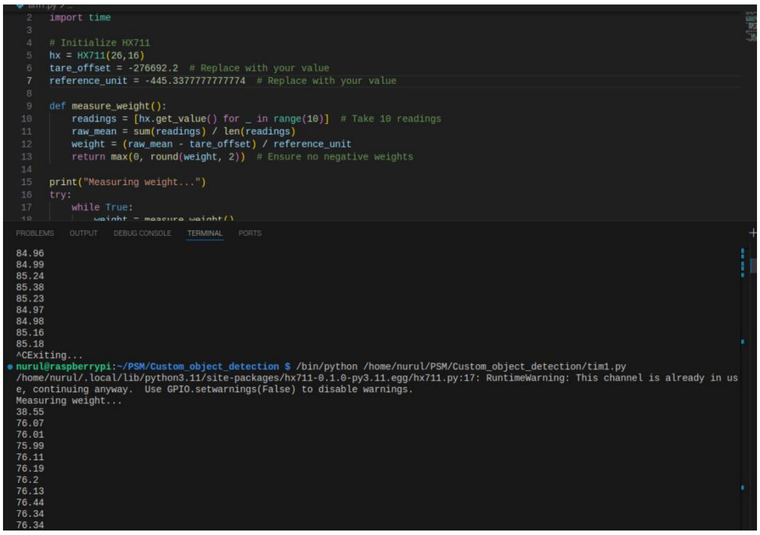


Figure 4. 17 The output of weight measurement of the product

The process of measuring the weight of products in the system. The system is designed to accurately weigh various products, including garlic, lime, chili, and potato using a calibrated load cell sensor. Each product is placed on the weighing platform, and its weight is recorded as shown in Appendix A.

Table 4. 5 presents the average values for 50 measurement data for each product. The slight variation in measured values across all items highlights the reliability and consistency of the weighing system, with minimal deviations from the raw weight values. These results validate the system’s ability to produce accurate and repeatable measurements for the tested items, demonstrating its suitability for precise weight determination in practical applications.

Table 4. 5 The average of each product

Product	Raw value (g)	Average (g)	Accuracy
Potato	86 g	85.79 g	99.75%
Garlic	76 g	76.16 g	99.79%
Lime	42 g	39.48 g	93.62%
Chili	23 g	22.83 g	99.25%

4.3 Summary

The convergence of a barcode scanner and an image processing capability into an automated self-billing counter using the Raspberry Pi demonstrated a compact solution for modern retail environments. The results highlighted how the project used the Pi Camera for image processing and user inputs through the keypad visual and auditory feedback. Further validation of the design was done through circuit simulations that revealed the seamless interactions between the various hardware components and the Raspberry Pi for accurate item recognition and billing. It also makes everyday life more convenient even for the users who benefit most from this technological advance, as going to the market and shopping will no longer involve adhering to long queues for checkout.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

The last chapter will provide a summary to conclude the overall project that has been carried out during the entire final year project while giving and providing some suitable suggestions for future work.

5.2 Conclusion

The main objective of this thesis was to develop an automated self-billing counter with barcode scanning and image processing on Raspberry Pi. Inspired by prior research on smart cart systems and automated self-checkout counters, this project adopted the Raspberry Pi in combination with Visual Studio Software to obtain the desired outcomes. The rationale for this selection was driven by the high cost associated with the existing project, necessitating the development of a more cost-effective alternative. The proposed system was designed to be user-friendly while ensuring high accuracy, precision, and enhanced overall performance. Upon successful completion, the automated self-billing counter with barcode scanning and image processing was developed at an estimated cost of RM 350.

The automated self-billing counter is designed to operate across two primary scenarios, with the first scenario focusing on the barcode scanning process. This scenario serves as a critical component of the system, ensuring accurate identification and registration of products. Five products with barcode labels were utilized during the testing phase to assess the reliability of the barcode scanning functionality. The testing process involved multiple trials to determine the system's consistency and accuracy in recognizing and processing barcodes under various conditions. The results demonstrated that the barcode scanning process achieved 100% reliability, with each product being accurately identified in every trial. This high level of reliability highlights the robustness and efficiency of the system in performing its intended functions.

The second scenario of the automated self-billing counter involves image processing functionality. This scenario was evaluated using four distinct products: garlic, chili, potato, and lime. The objective was to assess the system's ability to accurately and precisely identify these products based on image processing techniques. Specifically, the potato achieved the highest accuracy at 92.09% and a precision of 88.09%. Garlic and chili both exhibited comparable performance, with accuracy values of 90.42% and 90.70% and precision levels of 83.30% for both. Lime recorded an accuracy of 90.61% and a precision of 79.07%.

The weighing process is seamlessly integrated with the image processing functionality, enabling the automated self-billing counter to identify and weigh products simultaneously. The weighing process utilizes a load cell sensor to measure the weight of products accurately. Each product was tested 50 times to evaluate the system's performance. The results demonstrated high levels of accuracy, with the average weight of the product closely matching the raw value. For instance, the potato recorded an accuracy of 99.75%, while garlic, chili, and lime achieved accuracies of 99.79%, 99.25%, and 93.62%, respectively. This finding indicates that the system's highly reliable and consistent weighing capability contributes to a seamless checkout process.

In conclusion, the development of the automated self-billing counter utilizing barcode scanning, image processing, and load cell sensor technology has successfully demonstrated its potential to revolutionize the retail checkout process. The system addresses key challenges such as long waiting times, shoplifting, and human errors while enhancing customer convenience and operational efficiency. With high levels of accuracy and precision achieved in barcode recognition, image-based product identification, and weight measurement, the project has validated the feasibility of integrating these technologies into a cost-effective and user-friendly solution. Additionally, the ability to send the bill directly to customers via WhatsApp ensures an efficient and eco-friendly approach to billing, further enhancing the user experience.

5.3 Potential of the project

The automated self-billing counter project, when made viable, has the power to cause a revolution in the retail business. This project, based on the technologies of Raspberry Pi along with processes of image and barcode scanning, ensures faster checkout with no delays and full satisfaction for the customers. It ensures that it is compact, and cost-effective in its design, and opens up the small and medium retailer field for its large-scale use. The enhanced scalability of this project also provides for integration into existing retail infrastructures, for real-time inventory management and data analytics functionality. It is believed that this innovation will improve operational efficiencies and make the shopping experience smooth and phenomenal, thereby standing as a key development in the future of retail automation.

5.4 Future Works

In the future, improvement in the automated self-billing counter results should focus on the following areas:

a) More advanced machine-learning algorithms can be applied to the system for object recognition and classification with greater precision, which can be learned over time and improved with the increase of inputs.
b) A new scanner that was upgraded with better capabilities to scan diversified formats at a faster pace and accurately, so that identification by the scanner of diversified productions will not be interrupted.
c) Use high-resolution cameras along with sophisticated image processing techniques, which further enable it to draw more differences between the lookalike items and, therefore, improve the accuracy of the item detection system.

REFERENCE

[1] R. Rahul, S. N. P. Saastha, P. S. Sai, M. S. Yashwanth, and R. Raffik, "Automated Smart Trolley System using RFID Technology," in *2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation, ICAECA 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICAECA56562.2023.10199259.

[2] *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, 2020.

[3] *Proceedings of the 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA 2020) : 5-7, November 2020*. IEEE, 2020.

[4] T. Sarala, Y. A. Sudha, K. V. Sindhu, C. H. Suryakiran, and B. N. Nithin, "Smart electronic trolley for shopping mall," *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018 - Proceedings*, pp. 2422–2427, 2018, doi: 10.1109/RTEICT42901.2018.9012466.

[5] V. Jabade, M. Talewar, S. Surve, S. Tadavi, and S. Tak, "Automatic Billing Trolley," in *Proceedings of the 1st IEEE International Conference on Networking and Communications 2023, ICNWC 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICNWC57852.2023.10127314.

[6] N. X. Jie, I. Farahana, and B. Kamsin, "Self-Checkout Service with RFID Technology in Supermarket," 2021.

[7] S. Umar, S. Twaibu, G. G. Ocen, L. Badru, A. Andrew, and M. Davis, "RFID-Based Automated Supermarket Self-Billing System," *East African Journal of Information Technology*, vol. 4, no. 1, pp. 7–15, Nov. 2021, doi: 10.37284/eajit.4.1.488.

[8] M. R. Krishna, "RFID-Powered Intelligent Shopping Cart : Streamlined Checkout and

Automated Billing," *2024 IEEE International Conference for Women in Innovation, Technology & Entrepreneurship (ICWITE)*, no. Icwite, pp. 345–348, 2024, doi: 10.1109/ICWITE59797.2024.10503373.

[9] B. Bukova, J. Tengler, and E. Brumercikova, "A model of the environmental burden of RFID technology in the Slovak Republic," *Sustainability (Switzerland)*, vol. 13, no. 7, Apr. 2021, doi: 10.3390/su13073684.

[10] S. Seo and S. S. Park, "Entry regulations with implementation lag: Evidence from convenience store markets in Korea," *Int J Ind Organ*, vol. 93, no. February, p. 103057, 2024, doi: 10.1016/j.ijindorg.2024.103057.

[11] Y. Sun, S. Q. Wang, Z. Q. Ou, S. Y. Chen, and S. L. Hsueh, "Using DFuzzy to Build Multi-attribute Decision-making Model for Chain Convenience Store Marketing," *Proceedings of the 6th IEEE Eurasian Conference on Educational Innovation 2023: Educational Innovations and Emerging Technologies, ECEI 2023*, pp. 325–329, 2023, doi: 10.1109/ECEI57668.2023.10105342.

[12] T. Kumar, "Hypermarkets Market Size & Share Analysis - Growth Trends & Forecasts (2025 - 2030) Source: https://www.mordorintelligence.com/industry-reports/hypermarkets-market."

[13] A. M. Shakir, Y. Mudhafar, and A. M. A. Al-Muqarm, "Modernizing The Shopping Experience: The Smart Shopping Cart," in *6th Iraqi International Conference on Engineering Technology and its Applications, IICETA 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 636–640. doi: 10.1109/IICETA57613.2023.10351269.

[14] N. Rondan, J. Fernandez-Palleiro, R. Salveraglio, M. Eugenia Rodriguez-Rimoldi, N. Ferro, and R. Sotelo, "Self-Checkout System Prototype for Point-of-Sale using Image Recognition with Deep Neural Networks," *2021 Ieee Urucon, Urucon 2021*, pp. 217–222, 2021, doi: 10.1109/URUCON53396.2021.9647049.

[15] Robert L Dilonardo, “Self-Checkout Reaches Critical Mass,” Jan. 02, 2024. Accessed: Apr. 24, 2024. [Online]. Available: https://losspreventionmedia.com/self-checkout-reaches-critical-mass

[16] M. Yokeshwaran and J. Murugachandravel, “Enhancement of business using e-commerce with point of sale,” *Int J Health Sci (Qassim)*, vol. 6, no. March, pp. 3250–3260, 2022, doi: 10.53730/ijhs.v6ns2.5811.

[17] M. Getsy *et al.*, “An Innovative Shopping System with GSM Based Automation for Physically Challenging and Old Age People,” in *Proceedings of 2023 3rd International Conference on Innovative Practices in Technology and Management, ICIPTM 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICIPTM57143.2023.10117580.

[18] T. Tamilselvi, G. Puthilibai, R. Kavina, I. Delhi Ganesh, J. Varsha, and S. Karthick Kumar, “Autonomous Billing Cart for Retail Store,” in *2023 Intelligent Computing and Control for Engineering and Business Systems, ICCEBS 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICCEBS58601.2023.10448643.

[19] Q. Guo, Y. Chen, Y. Yao, T. Zhang, and J. Ma, “A Real-Time Chinese Food Auto Billing System Based on Instance Segmentation,” in *2023 IEEE Region 10 Symposium, TENSYMP 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/TENSYMP55890.2023.10223619.

[20] V. P. Chaudhari, R. V. Chaudhari, A. S. Burgul, and S. A. Jain, “Smart Self-Checkout System for Supermarkets,” *3rd IEEE International Conference on Technology, Engineering, Management for Societal Impact using Marketing, Entrepreneurship and Talent, TEMSMET 2023*, pp. 1–6, 2023, doi: 10.1109/TEMSMET56707.2023.10150048.

[21] B. Kumar Yadav, A. Burman, A. Mahato, M. Choudhary, and A. Kundu, “Smart Cart: A Distributed Framework,” *2020 IEEE International Conference for Convergence in Engineering, ICCE 2020 - Proceedings*, pp. 210–213, 2020, doi:

10.1109/ICCE50343.2020.9290605.

[22] M. Jaishree, K. R. Lakshmi Prabha, S. Jeyaprabha, and K. Mohan, "Smart Shopping Trolley Using IOT," in *2021 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021*, Institute of Electrical and Electronics Engineers Inc., Mar. 2021, pp. 793–796. doi: 10.1109/ICACCS51430.2021.9441786.

[23] A. Titus, U. S. Mahesh, G. Nandu Krishnan, and M. Francis, "DeepWeight: A Semi-Autonomous Food Weight Billing System using Deep Learning," in *2023 International Conference on Control, Communication and Computing, ICCC 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICCC57789.2023.10165428.

[24] N. D. Perez, J. Villaverde, W. Cereneo, A. Custodio, S. B. Conde, and L. Bulawan, "Design and Development of a Raspberry Pi-Based Shopping Cart Following Robot through Computer Vision and Object Tracking Method," in *2022 IEEE Region 10 Symposium, TENSYMP 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/TENSYMP54529.2022.9864520.

[25] A. A. Santoso Gunawan *et al.*, "Line Follower Smart Trolley System V2 using RFID," in *Proceedings of 2021 1st International Conference on Computer Science and Artificial Intelligence, ICCSAI 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 17–21. doi: 10.1109/ICCSAI53272.2021.9609710.

[26] *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, 2020.

[27] D. K. Burada, P. V. Charan, G. J. Rani, K. S. Nikhil, S. K. Dargar, and A. Gupta, "Low-Cost Automatic Purchase Cart for Hassle-free Uncrowded Shopping in Indian Supermarkets," in *Proceedings - 2022 2nd International Conference on Innovative Sustainable Computational Technologies, CISCT 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/CISCT55310.2022.10046503.

[28] T. Sangkharat and J. La-Or, "Application of Smart Phone for Industrial Barcode Scanner," in *2021 7th International Conference on Engineering, Applied Sciences and Technology, ICEAST 2021 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021, pp. 9–12. doi: 10.1109/ICEAST52143.2021.9426288.

[29] Y. Zhang and X. Zheng, "Development of Image Processing Based on Deep Learning Algorithm," in *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers, IPEC 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1226–1228. doi: 10.1109/IPEC54454.2022.9777479.

[30] B. Yang and X. Chen, "Face Recognition Implementation Based on Image Processing Techniques," in *Proceedings of 2024 2nd International Conference on Signal Processing and Intelligent Computing, SPIC 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1105–1109. doi: 10.1109/SPIC62469.2024.10691607.

[31] M. J. Islam Jubaid *et al.*, "Design And Construction of a Smart Restaurant Service Using Conveyor Belt," in *IEEE International Conference on Advances in Electronics, Communication, Computing and Intelligent Information Systems, ICAECIS 2023 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 265–270. doi: 10.1109/ICAECIS58353.2023.10170087.

APPENDICES

Appendix A Data measurement of tested items.

a) Potato

Items	Raw value (grams)	Measured value (grams)
Potato	86	85.79
	86	86.61
	86	86.49
	86	85.97
	86	85.72
	86	85.56
	86	85.61
	86	85.54
	86	85.39
	86	85.43
	86	85.54
	86	85.5
	86	85.52
	86	85.87
	86	85.78
	86	85.58
	86	85.5
	86	85.41
	86	85.37
	86	85.46
	86	85.51
	86	85.34
	86	85.23
	86	85.24
	86	85.28
	86	85.18
	86	85.34
	86	85.43
	86	85.46
	86	85.41
	86	85.5
	86	85.51
	86	85.52
	86	85.43
	86	85.43
	86	85.39
	86	85.39

	86	85.31
	86	85.44
	86	85.57
	86	85.66
	86	85.67
	86	85.67
	86	85.65
	86	85.45
	86	85.45
	86	85.41
	86	85.36
	86	85.26
	86	85.33

b) Garlic

Items	Raw value (grams)	Measured value (grams)
Garlic	76	76.07
	76	76.01
	76	75.99
	76	76.11
	76	76.19
	76	76.2
	76	76.13
	76	76.44
	76	76.34
	76	76.34
	76	76.27
	76	76.33
	76	76.3
	76	76.58
	76	76.46
	76	76.31
	76	76.17
	76	76.03
	76	76.16
	76	76.11
	76	76.19
	76	76.22
	76	76.26
	76	76.22
	76	76.2
	76	76.14
	76	76.1

	76	76.03
	76	76.1
	76	76.09
	76	76.06
	76	76
	76	76.04
	76	76.02
	76	75.97
	76	75.97
	76	76
	76	76.03
	76	76.03
	76	76.08
	76	76.16
	76	76.19
	76	76.1
	76	76.06
	76	76.02
	76	76.04
	76	76.15
	76	76.26
	76	76.29
	76	76.3

c) Lime

Items	Raw value (grams)	Measured value (grams)
Lime	42	39.48
	42	39.42
	42	39.54
	42	39.54
	42	39.18
	42	39.46
	42	39.6
	42	39.61
	42	39.71
	42	39.87
	42	40.03
	42	40.19
	42	40.37
	42	40.78
	42	40.53
	42	40.58

	42	40.47
	42	40.76
	42	40.52
	42	40.43
	42	40.36
	42	40.4
	42	40.26
	42	40.17
	42	40.22
	42	40.21
	42	39.99
	42	39.8
	42	40.03
	42	40.12
	42	39.99
	42	39.49
	42	39.75
	42	39.5
	42	39.07
	42	39.27
	42	39.41
	42	39.56
	42	39.7
	42	39.74
	42	39.84
	42	39.77
	42	39.79
	42	39.86
	42	39.78
	42	39.86
	42	39.86
	42	39.61
	42	39.71
	42	39.41

d) Chili

Items	Raw value (grams)	Measured value (grams)
Chili	23	22.65
	23	22.41
	23	21.86
	23	21.63
	23	21.9

	23	21.62
	23	21.75
	23	22.8
	23	22.86
	23	23.21
	23	22.72
	23	23.42
	23	22.75
	23	22.65
	23	22.93
	23	22.94
	23	22.88
	23	22.68
	23	22.67
	23	22.75
	23	22.4
	23	22.27
	23	22.92
	23	23.35
	23	23.28
	23	22.95
	23	22.91
	23	22.56
	23	22.37
	23	22.83
	23	23.17
	23	23.41
	23	23.03
	23	22.54
	23	22.61
	23	22.58
	23	23.06
	23	22.74
	23	22.92
	23	22.83
	23	22.74
	23	23.61
	23	23.87
	23	23.51
	23	23.62
	23	23.34
	23	23.32
	23	23.46
	23	23.38
	23	23.03

23% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Crossref database
- Crossref posted content database

Match Groups

- **223** Not Cited or Quoted 21%
  Matches with neither in-text citation nor quotation marks
- **18** Missing Quotations 1%
  Matches that are still very similar to source material
- **10** Missing Citation 1%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

15% Internet sources
7% Publications
21% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.