بسم الله الرحمن الرحيم

قال الله تعالى

{ يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ }

سورة المجادلة آية رقم (١١)

TOWARD DEVICE-ASSISTED IDENTIFICATION OF GROCERY STORE SECTIONS AND ITEMS FOR THE VISUALLY IMPAIRED

By Dalia Essam Attas

A thesis submitted for the requirements of the degree of Master of Science / Computer Science

Supervised by

Dr. Wadee Halabi

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY
JEDDAH - SAUDI ARABIA
Ramadan 1436 H - Jun 2015 G

ACKNOWLEDGEMENTS

First of all, I am thankful to Allah for good health, wellbeing, and being on my side all the time that were necessary to achieve this journey successfully.

I wish to express my serious thanks to my supervisor Dr. Wadee, for his valuable support and scientific recommendations to improve my learning and writing skills.

I am also grateful to my parents, who teach me all my life to never give up and be optimistic.

I extend my thanks to my dear husband, for his guides and endless believes about my capabilities in all my rough situations.

I take this opportunity to express gratitude to my sweet sisters, who gave me the encouragements and supports.

Finally my thanks to my beloved sweet kids, for being patient to handle the time when I was away from them.

TOWARD DEVICE-ASSISTED IDENTIFICATION OF GROCERY STORE SECTIONS AND ITEMS FOR THE VISUALLY IMPAIRED

Dalia Essam Attas

ABSTRACT

There are number of visually impaired persons worldwide who need assistance in daily life tasks such as grocery shopping. The visually impaired persons usually need the assistance in grocery shopping from other persons or from an assisting tools. In order to preserve the visually impaired privacy and independency, a system should be constructed as grocery shopping assistant.

There are number of systems developed to assist the visually impaired in grocery shopping. The developed systems require massive work from the users to operate the system devices. Furthermore, the systems require wireless connections and products database to obtain the products information. Here comes the need for a system that assist the visually impaired person in grocery shopping with a shopping cart without any additional devices. Moreover, the system should use object recognition algorithms instead of wireless connections and database to recognize products.

In the thesis, a system was created to solve the problem of assisting the visually impaired in grocery shopping. The system workflow consists of three stages: i) Announcing the aisle category name to the user, ii) Finding the user desired product on the shelf, iii) Guiding the user to the product location. The thesis recommends implementing a shopping cart consists of three cameras installed vertically on one side of the cart. Furthermore, a comparison between two object recognition algorithms to recognize products on the aisle shelves was conducted. Additionally, a multimodal system was created to fuse the results of the used object recognition algorithms. The results showed that the fusion performed better results than the usage of each algorithm separately.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

The World Health Organization stated that 285 million people worldwide were visually impaired (39 million blind and 246 with low vision), from which 82% are in their 50s or above [1]. The visually impaired people in the age of 50 and above are considered to require assistance for the daily routine tasks such as grocery shopping.

The visually impaired people can use grocery shopping in only three ways. Either by asking someone to do their shopping or by employing online grocery shopping stores or by asking someone to join them at the grocery store. Neither one of the three ways gives the visually impaired the independence in shopping and choosing what they like.

The more accessible way for the visually impaired to shop in the grocery store is to go the store themselves. That gives them the freedom to select the products without the need for a previous task such as filling an electronic shopping list. Conversely, there are many obstacles that face the visually impaired in attending the grocery store such as the lack of assisting tools that can help them performing such trivial tasks. Furthermore, there are no resources in locating items that the visually impaired desire to purchase.

Guide dogs and white canes are the most known supporting tools that the visually impaired people use in guidance. Although the main function of the guide dogs is to help in navigating through the routes such as walking through the aisles in the grocery store, the guide dogs are incapable of differentiating between the different sections of a grocery store or identifying particular items. A guide dog may be capable of locating items along a route by memorization due to repetition and routine. Conversely, a guide dog is rendered useless for identifying items when a store is remodeled or undergoes any changes. Obviously, the white cane cannot achieve any more than the guide dog in the discussed scenarios.

There are several institutions active in research and development for systems assisting the visually impaired persons to perform individually grocery shopping, e.g., shopping in supermarkets. Some of these efforts deal with assisting the person to navigate inside the supermarket while others focus on assisting the person in locating the required product. The wearable devices gained the extra attention in assisting the visually impaired persons. Some systems require physical work from the user. Additionally, some systems rely on wireless connections and databases to recognize products. Here came the importance to develop a system that not exhaust the user. Also, the system should rely on object recognition algorithms to detect products without referring to other materials.

There are different object recognition algorithms that can be used to detect objects using image features. Usually, the grocery products images show the text written on the products that can be recognized as text features and the overall shape of the products that can be recognized by visual features. The visual features can be used for recognition by tools such as feature matching algorithms. on the other hand, text features can be used to recognize text written on products through means of optical character recognition (OCR) algorithms. Due to the number of image features that can be used to recognize the object, the current thesis compared the visual features recognition algorithms with the textual features recognition algorithms. Also, the two algorithms fused together in order to create a multimodal system. Accordingly, there are number of fusion levels to create a multimodal system. We described each level and the level in use. Additionally, we evaluated each result from the object recognition algorithms using the confusion matrix and the performance rates.

The main objectives of the thesis is to develop a system that can perform three important tasks: i) Announcing the aisle category name to the user, ii) Finding the user desired product on the shelf, iii) Guiding the user to the product location. In order to recognize the products (second task), we compared the visual features recognition and textual features recognition. The algorithms implemented under 1550 product images collected from a dataset on web. The visual features are compared along with product images on a dataset for recognition and the textual features are compared with a name of a product specified by the user. Also, a multimodal system consisting of the fusion between the visual feature recognition and textual feature recognition via an enhanced fusion level procedure was developed.

The thesis discussed the performance rates of each algorithm results in the context of minimum and maximum rates. Each unfamiliar results is detailed and explained. Furthermore, the fusion results are presented in comparison with the ones before fusion. Also, a comparison will be implemented between the study results and the background systems.

1.1 Research Aim and Objectives

The general aim of this work is to create a functional efficient system, easy to use. This is realized by i) solving some issues that can disrupt the user while using the system and ii) allowing the extension of the system usage to any grocery store by limiting the data sources. The proposed objective is to develop a software system that can achieve the following:

1. The system can work in any grocery store without any further implementation.
2. The visually impaired people can shop in the grocery store like any shopper using only a specially equipped shopping cart.
3. The system announces to the user the name of the products category (grocery store section) in the aisle by speech.
4. The system retrieves to the user desired product location in the aisle shelves.

One novel aspect of the study is the comparison of two object recognition algorithms: OCR (2.3.2) and IPM (2.3.1) on grocery store products dataset. The comparison is aimed to justify which method is applicable for implementation and to show the result of both methods. Another novel aspect is represented by the fuse between the IPM and OCR and its performance assessment versus each algorithm alone.

1.2 Thesis Overview

In Chapter one, the general aspects related to the problem of grocery shopping by the visually impaired persons is presented. This chapter also contains the main objectives of the thesis and a general overview.

Chapter two described the background systems used for assisting the visually impaired in grocery shopping. Furthermore, a background about the object recognition algorithms and system examples is explained. The most known evaluation method for object recognition is described. The chapter also contained the multimodal fusion levels and system examples.

Chapter three described the system environment and workflow. The product recognition phase is explained according to the IPM and OCR algorithms. The dataset used in the system is counted and explained. The evaluation method is explained according to the algorithms used.

In Chapter four, results of the algorithms are shown according to the evaluation method using graphs. Each evaluation method is clarified with the method procedure and examples. Also, the fusion results are showed using graphs along with the procedure and an example.

The following chapter contained the results discussion. Each result is discussed according to the minimum and maximum results. Additionally, the unfamiliar results are explained.

Chapter six contained the conclusions of the thesis, specifying the main concepts mentioned in the thesis, empirical findings, and limitations. Upon each finding, we recommended the future work.

Chapter 2

Literature Review

There are several institutions active in research and development for systems assisting the visually impaired persons to individually perform grocery shopping, e.g., shopping in supermarkets. Some of these efforts deal with assisting the person to navigate inside the supermarket while others focus on assisting the person in locating the required product.

The systems navigate the user inside the store using electrical devices or using computer vision algorithms. The electrical devices identify the products using Radio-Frequency Identification (RFID) tags or Barcodes. On the other hand, the computer vision algorithms are employed for product identification using a camera that captures real scenes from the store. Some of the systems rely on a database containing the store products or map. Other systems use a server database that collect different products from different brands.

The next section (2.1) will discuss the past systems in assisting the visually impaired persons. Section (2.2) will cover a comparison between the previous systems. Section (2.3) will explain Object Recognition methods including image feature matching and Optical Character Recognition (OCR). Section (2.4) introduces the performance evaluation techniques that will be used to evaluate the results. Section (2.5) will discuss the multimodal concept and the levels of the fusion. Finally, Section (2.6) concludes the chapter with the main concepts and methods.

2.1 Background Systems for assisting visually impaired persons

2.1.1 RoboCart

One of the leading efforts in this domain is the RoboCart [2]. Robocart was developed through a collaboration between the Computer Science Assistive Technology Laboratory (CSATL) at the Utah State University (USU) and the Center for Persons with Disabilities (CPD). The RoboCart, shown in Figure (2.1), is a robot that assists the visually impaired by helping the user to navigate the grocery store to purchase the desired items. RFID tags were attached to the store shelves to help localize the products in the store. The RoboCart hardware consists of a robotic platform and a navigation system. The navigation system includes a laptop, hand-held keypad, laser range finder, RFID reader, and antenna.

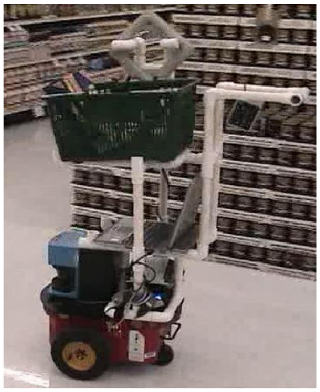


Figure 2.1: RoboCart [2]

RoboCart was deployed using three components: the user interface (UI), path planner, and behavior manager. The UI took the user distention selection from a hand-held keypad as an input. The path planner turns the UI input as a goal to generate a path from the current point to the end point. The behavior manager assists in observing RoboCart overall global state. The system represents the environment of the grocery store as a connectivity graph where the nodes are the RFID tags, and the edges represent the path from one tag to another tag. The path planner uses the

Breadth-First Search algorithm (BFS) in order to discover the shortest way from the starting tag to the destination tag; both located in the connectivity graph.

According to the RoboCart developers, the system provides additional assistance but does not replace the white cane and guide dogs. On the other hand, the white cane and guide dogs cannot individually assist the user in navigation because of to their lack of information about the environment and the user's intentions. The RFID tags are inexpensive, reliable, maintainable and do not require external power supplies. Some failure may exist in the RFID reading. The small size of the RFID tags is helpful in not annoying the customers and the workers. Some supermarket owners will find it expensive to apply and maintain the RFID tags according to the number of adjustments that will accrue in the supermarket.

RoboCart can assist the user to reach the grocery store sections, but cannot help the user in retrieving a particular product from the store shelves. Meanwhile, RoboCart sometimes does not know when to execute a U-turn at the end of the aisle. Also, the system was deployed and tested in a grocery store but not evaluated by visually impaired customers [2].

According to [3], the RoboCart participants claimed that the system lacks to be independent. In 2008, a new concept was introduced to help in providing an independent shopping. The system presents the shopper with an interface of haptic and locomotor spaces in the grocery stores. The haptic space is a space around the user that can be sensed by touch without moving his body. On the other hand, the locomotor space is a space that requires moving from space to space. In addition, the system assists the user in navigation and product retrieval.

The haptic module represented using a modified barcode reader that present natural alignment with the shelves. After the user scans the barcode, the system announces the name of the product to the user. Through voice instructions, the user is guided to the aisle in order to get the required item. Then, the user retrieves the required item from the shelves independently. The system uses laser-based Monte Carlo Markov localization (MCL) for navigation in the locomotor space. In order to receive reliable MCL results, the system switches the store floor to an RFID surface. The RFID surface is an RFID mat which contain internal RFID tags.

Three product selection interfaces were tested: browsing, typing, and speech. In order to find the user desired product, the browsing interface aims to browse a product category to the user. The typing and speech interfaces were build based on information retrieval method. In the typing interface, the user must type the search name of the product with a numeric keypad. In the speech interface, the system applies speech recognition to recognize the product. A product repository that contains a number of the grocery store products is used. The typing interface searches for a partial product name in the product repository by the word prediction tree. The word prediction tree returns to the user all the predicted options. In the case of the speech interface, the user is required to speak one product name at a time.

The new approach has a series of disadvantages, the most significant being that the user needs to scan all the barcodes in order to find the required barcode. There is a situation that he may not find the product at all in case he misses scanning the product barcode. The results of experimenting the system on the users showed that, over time, the user can retrieve the desired product after customizing himself in the haptic space [3].

The developers of RoboCart outspread the project to involve a new system called ShopTalk [4]. ShopTalk was proposed in 2009 as an extension of the locomotor and haptic space to comprise the search space. The difference between the locomotor space, the haptic space, and the search space is that the locomotion space is used to help the user to get to the requested store section. During the search space, the system assists the user to move into the aisle to identify the location of the required product. The haptic space is a space around the requested product that requires a physical motion from the user to catch the product.

ShopTalk is a wearable system developed to assist the visually impaired people to find a product on the grocery store shelves. It uses speech directions and a map of the grocery store to find the aisle of the required item. The user is directed in the aisle using a locomotor space topological map and a Barcode Connectivity Matrix (BCM). A topological map is a directed graph with nodes of a decision points (ex. the store entrance, the aisle entrance). The topological map edges are regarded as the route directions. The authors built the BCM from the store inventory database that connects each product barcode with product information.

ShopTalk hardware (Figure (2.2)) consists of the computational unit, numeric keypad, wireless barcode scanner, USB hub to connect components, backpack to carry parts, and headphone. The testing results showed that the users were able of navigating to the required aisle with the assistance of the verbal routes. Also, the users were able to find all the required products using BCM and barcode scans. The concluded success rate of retrieving the required products was 100%. The main advantage of the system is that it does not use any tool from the store, which leads to lower cost and maintenance. The only software utilized is the topological map [4].

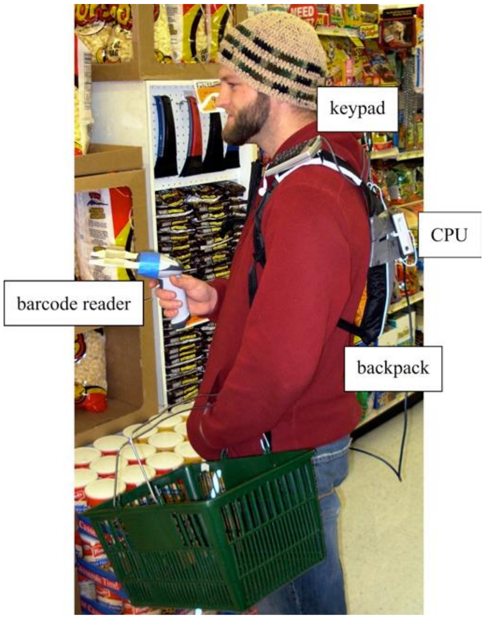


Figure 2.2: ShopTalk [4]

In 2010, ShopTalk introduced a new generation of the system called ShopMobile [5]. ShopMobile aimed to replace the design of the ShopTalk with a mobile phone and Bluetooth-pen barcode scanner for scanning the product barcode. An advance version of ShopMobile was introduced to replace the barcode scanner with a computer vision system recognizes the barcode on the shelves using a mobile phone camera [5].

2.1.2 Trinetra

In 2006, a system called Trinetra [6] was developed by Carnegie Mellon University (CMU). The system assists the visually impaired in shopping at grocery stores independently and cost-effectively using COTS (Commercial off the shelf) products. The purpose of the system was to help the users in identifying the desired product. An additional goal is to differentiate between the stores products that are on the same shelf. The system used both RFID tags and UPC (Universal Product Code) barcode for product identification. The authors claimed that the grocery store constructors do not commonly use RFID tags in the stores. For that reason, the UPC was a preferred option. The users will need to seek for help in determining the required aisle and shelf.

UPC barcode is tagged in every product, in the grocery store, by the product manufacturer. There is an online database that provides the integration between the UPC and their corresponding description of the product.

As shown in Figure (2.3), the system consists of a barcode scanner, smart cell phone, and a Bluetooth headset. When the user scans a grocery product, the barcode scanner will send the barcode to the phone using Bluetooth connection. The phone contains a Symbian module to start a Hypertext Transfer Protocol (HTTP) communication with the UPC online database. Then, the database will send the product information as text to the phone. Finally, the phone will announce the product information using text to speech software.

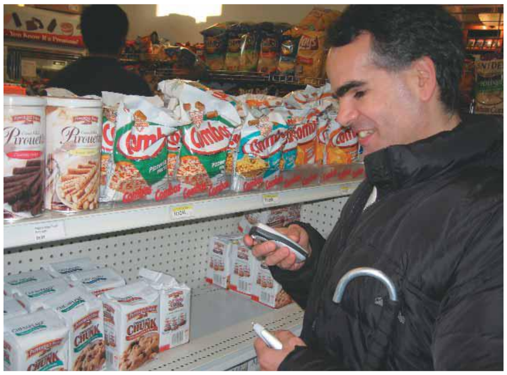


Figure 2.3: Trinetra [7]

According to the system authors, the system has a number of advantages such as cost effectiveness, independently shopping, and portability. Trinetra will not require doing any modification in the store on account of the barcode solution. A blind person was helpful to evaluate the system. In contrast, the barcode solution considered to be inconvenient because the user will need help in locating the required aisle and shelf [6] [7].

The portability advantage may conflict the user such that he will need to hold the phone and barcode scanner, in the same time holding the product or carry on a basket. Another drawback consisted in the fact that the system did not assist the user in recognizing a particular product that he wants to purchase. In addition, there is the problem of the UPC online database, which may not include all the products in the store.

2.1.3 GroZi

The California Institute of Telecommunications and Information Technology and Computer Science and Engineering department at the University of California, San Diego collaborated by the year 2007 to develop a system called GroZi [8]. This is based on a previous attempt aimed to construct a hardware called MoZi box which downloads the shopping list to the device. The MoZi box consists of Servo motors and camera. The Servo motors assist the user in navigating the aisle while the camera scans the words and compares each word with the words in the shopping list. The MoZi box was only an idea, and the authors did not implement the device.

GroZi used computer vision techniques to assist the visually impaired in locating any item in any specified area. Once the system locates the user required item, feedback is sent to the user using touch sensors. GroZi searches for the required items using a shopping list that is filled by the user before attending the supermarket. The system consists of three main parts.

i) A website which contains a shopping list for the visually impaired persons.

ii) Computer vision software is used to assist in identifying the camera views that include grocery items by object recognition algorithms.

iii) A portable device that memorizes the contents of the shopping list. It runs the computer vision software and then extracts feedback to guide the user to the specific item.

As shown in Figure (2.4), GroZi consists of a glove containing vibrating motors sensible of the four directions, a camera placed above the glove, Bluetooth headset for feedback, and a battery pack placed on a belt.

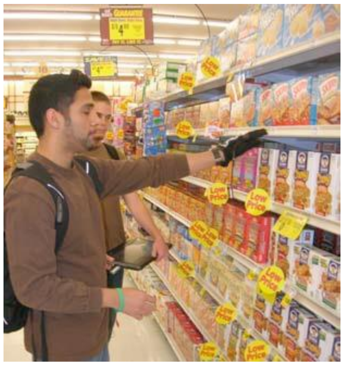


Figure 2.4: GroZi [8]

In order to test the system, the authors replaced the computer vision software with a Remote Sighted Guide (RSG). The RSG contains a person viewing what the camera is showing and control box aimed to provide audio and haptic feedback. When a person in the RSG displays the camera using a laptop, he will guide the user to the direction of the desired product by Bluetooth headset and the control box. When the user enters a store aisle, the camera on the glove is positioned to read the store sections sign. The name of the section is returned to the user using the headset. The user will need to change the position of the camera in order to scan the items on the shelf. The user will direct the camera to the products in both of the sides of the aisles slowly. The RSG will start determining whether the user shopping items in the list is available in the camera view. If RSG found the item, it would return to the user haptic feedback. It the item was not found, the user continues walking until the RSG finds the item.

According to the GroZi authors, the system device is easy to use, pleasurable to the user, and nonintrusive. The RSG model is efficient and durable. The system was tested according to the usability of the device. The experiments of the device showed that, it was hard for the user to find the items primarily. After the user had adapted to the system, he was able to find the items. The scope of the system does not include a solution to the portability issue such that if the user wants to hold the basket, cane or item [8].

GroZi only specifies to the user the aisle section and the location of the requested product on the shopping list. The user is not be able to purchase any item that is not on the shopping list. Furthermore, the RSG technique is not convenient because it cannot work without human control.

In 2011 [9], GroZi system introduced a new stage of development. The new system contained an Android application that can identify cereal box images and announce the name of the cereal to the user as shown in Figure (2.5). The authors thought that the new system would again a positive advantage such portability and flexibility.

The system works in two steps: image recognition and event handling. The image recognition step aims to recognize the products in the grocery store and it starts by collecting a number of images in a library and then extracts the image features through image recognition algorithms. When the application runs, the system computes the video frame features and matches them with the library images. If the application found a match, then the image is recognized. The event handling step uses verbal announcements, the system supplying the name of the products to the user.

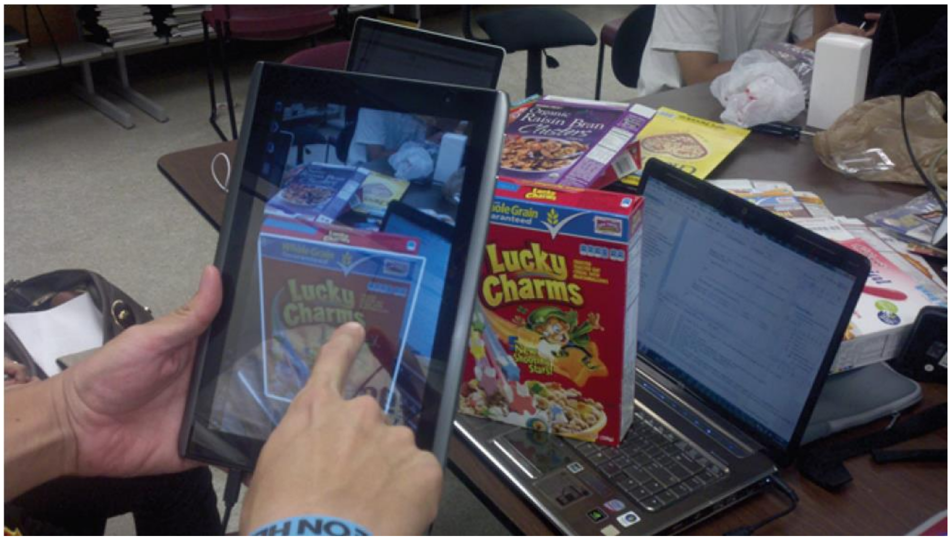


Figure 2.5: GroZi Android Application [9]

The main disadvantage of this system is that the users need to manually select the product from the tablet screen to announce the name of the product [9]. Furthermore, the system recognizes only obtained products from the library of the system.

In 2012 [10], GroZi was enhanced and the disadvantage of the previous model overcome by adding an auto detection feature. If the system recognized the product, then the product name is spoken to the user. The advantage of applying the auto detection feature is to free one of the user's hands to hold the grocery product while holding the tablet with the other hand. This model is more portable than the previous one, less error, and user-friendly. On the other hand, the authors did not repair the problem of recognizing only the products in the library [10].

2.1.4 Wearable Wireless RFID System for Accessible Shopping Environments

Sreekar Krishna and others created a system called "Wearable Wireless RFID System for Accessible Shopping Environments" [11] in 2008. The system aims to navigate the visually impaired persons in grocery shopping by retrieving the product information. The system first part uses the RFID tags in order to identify the products. The second part recognizes the product using a centralized store server. The two parts are related through a Transmission Control Protocol/ Internet Protocol (TCP/IP) interface. Figure (2.6) shows the system scenario.

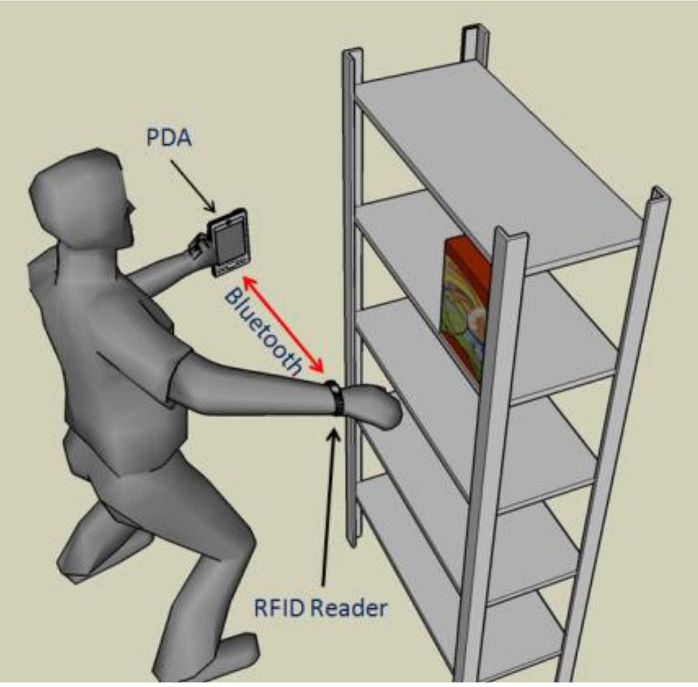


Figure 2.6: wearable wireless RFID system for accessible shopping environments [11]

The system consists of an RFID reader built into a wearable device, PDA with speech software, store server containing a database for the store products and an interface between the database and the wearable device.

When the user passes in front an RFID tag, the RFID reader picks up the tag ID attached to each product. Using Bluetooth, the tag ID is sent to the PDA which uses it to start a Wi-Fi connection with the store server. The interface in the store server compares the tag ID in the database to retrieve the product information. The database holds information about each product in the store, in which aisle, section, and shelf. This retrieved information is sent to the PDA in order to be announced to the user.

The system was experimented according to the RFID tags detection. The authors ran the tests under a number of circumstances, such as the time of the delay between reading a tag and announcing the information to the user. The results showed that the material of the product does not influence the performance of the tag reader. The users stated that the delay time from scanning the product until announcing the product information was considered not long. The system can considered a reliable tool for navigation because it can understand the location of the products and contents of each aisle [11].

As mentioned previously, the main disadvantage of the RFID tags is that they require many modifications in the store. Also, the store server may need a lot of data to be collected. Furthermore, the PDA devices are not suitable for the visually impaired because the user needs to catch the PDA at the same time he is holding a basket or moving a cart. When the user passes a product he will hear the product information whether he wants the product or not, so there is no product search according to the user order.

2.1.5 BlindShopping

By the year 2011, a system called BlindShopping [12] was developed to customize mobile techniques that assist the visually impaired at grocery shopping. The authors consider the system to have a minimal cost and easily deployable. It consists of three components: navigation, product recognition, and management. The navigation system is responsible for guiding the user in the grocery store using verbal directions through headphone. It consists of a white cane holding RFID reader, RFID tags attached to the supermarket floor as shown in Figure (2.7) and an application for the Smartphone. The application receives the RFID readings transmitted by the Bluetooth to extract the verbal directions of the RFID tag assigned to the product.

After the user arrives at the required section, the product recognition hints the user to direct the phone camera to the QR (Quick Response) code or UPC (Universal Product Code). The products or the products shelve contain the code as shown in Figure (2.7). The smartphone camera will recognize the code and announce the product information to the user. In the system management, the web page was developed for managing the system RFID and QR codes. It enrolls each RFID tag with the matching QR codes implemented in the products or store sections.

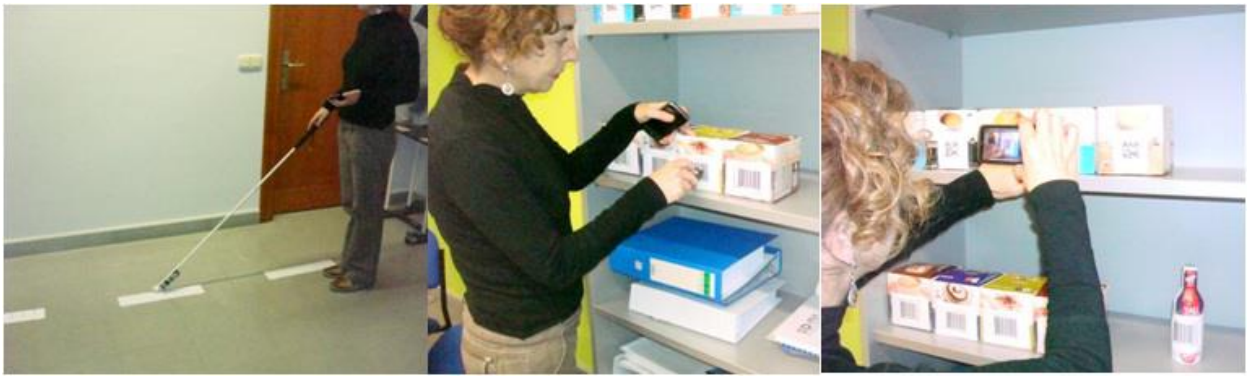


Figure 2.7: BlindShopping [12]

The system hardware was implemented using an RFID Bluetooth reader to read the RFID tags on the floor of the supermarket, and then send them to Java Bluetooth application to an Android phone carried by the user. The user can apply for an action using a system application in the Android phone by gesture or voice command. The navigation is activated by announcing the word location or by drawing the letter L on the phone screen. The product recognition system is activated by speaking the word Product or by drawing the letter P on the phones screen. A server is used for obtaining the business logic and the BlindShopping data. The server in the deployment is be joined with the supermarket inventory system.

After testing the system, the users found that the navigational system was smooth, and the verbal commands were useful. Also, the users claimed that pointing the camera to the QR codes are much plausible than pointing to the barcodes because QR codes are much earlier and dependable [12].

The system was considered to be high cost and rough to deploy due to the number of equipment that need to deploy in the supermarket such as the RFID tags on the floor, the QR codes or barcodes on the aisle shelves. Furthermore, the system will consume all the user physical power such that he needs to use his cane and sweep the phone camera over products continuously. There is no retrieval of products according to the user intentions; the user needs to sweep the camera on all the shelves until his desire product found.

2.2 Background Systems Comparison

Table (2.1) shows different features and methods used in grocery shopping systems for visually impaired people. Table (2.2) shows a comparison between the mentioned systems. The comparison discusses the advantages and disadvantages of the systems. Furthermore, the comparison is done according to the following shopping scenario:

1. The user is navigating inside the grocery store. (Store Navigation)
2. The user reaches the requested aisle. (Reach Requested Aisle)
3. The user search for the requested product. (Requested Product Search)
4. The user reaches the requested product. (Reach Requested Product)
5. The user retrieves the requested product. (Retrieve Requested Product)
6. The user is navigating to reach the cashier. (Cashier Navigation)

Table 2.1: Systems Features and Methods Comparison

System	Features	Method or Algorithm
RoboCart 2004 [2]	Navigation	RFID tag connectivity graph, Breadth first search algorithm (BFS)
RoboCart 2008 [3]	Navigation and Product Recognition	Monte Carlo Markov localization (MCL) and word prediction tree
ShopTalk [4]	Navigation and Product Recognition	Topological map and Barcode Connectivity Matrix (BCM)
ShopMobile [5]	Wireless	Bluetooth connection
Trinetra [6] [7]	Wireless	Bluetooth and HTTP connection
GroZi 2007 [8]	Wireless	Bluetooth connection
GroZi 2011and 2012 [9] [10]	Product Recognition	Image Recognition Algorithms
Wearable Wireless RFID System for Accessible Shopping Environments [11]	Wireless	TCP/IP and Wi-Fi
BlindShopping [12]	Wireless	Bluetooth connection

Table 2.2: Systems Advantage and Disadvantage Comparison

Systems	Store Navigation	Reach Requested Aisle	Requested Product Search	Reach Requested Product	Retrieve Requested Product	Cashier Navigation	Advantages	Disadvantages
RoboCart 2004 [2]	√	√	×	×	×	√	• Navigate the user in the store • RFID tags are cheap price, reliable, maintainable, small size and do not require outer power supply	• Cannot replace white cane or guide dogs. • System tested but not evaluated • RFID readings may fail • Sometimes cannot execute U-turn
RoboCart 2008 [3]	√	√	√	×	×	√	• The user can navigate inside the store. The user can reach the aisle of the requested product	• The user needs to scan all the barcodes in order to find the required barcode. • The user may not find the product at all in case he misses scanning the product barcode. The user can retrieve the desired product after customizing himself in the haptic space over time.
ShopTalk [4]	√	√	√	√	√	√	• The users were capable of navigating to the required aisle with the assistant of the verbal routes • The users were able to find all the required products using BCM and barcode scans • The concluded success rate of retrieving the required products was 100% The system design does not use any tool from the store and the only software used in the system is the topological map	• The topological map requires earlier knowledge of the store map and the interest points • The store may do not have a store inventory database • The shelves under the products may do not have a barcode for each product • The shoppers may not locate the products above their corresponding barcodes The hardware equipment may be conflicting to the user

ShopMobile [5]	√	√	√	√	√	√	• The system aims to achieve the portability goal by using a mobile phone and Bluetooth-pen barcode scanner for scanning the product barcode	• The system can conflict the user in the shopping
Trinetra [6] [7]	×	×	√	×	×	×	• Recognize products and different between products • Cost effective, independent shopping, and portability • Does not require modification in store	• User will need help in locating the required aisle and shelve • UPC database may not include all the store products • The portability feature may conflict the user in holding different objects
GroZi 2007 [8]	×	√	√	√	√	×	• The system assists the user in retrieving the requested products from the store shelves • The device is easy to use, pleasurable, and nonintrusive • The RSG model was effective and durable	• The experiments of the device showed that; it was hard for the user to find the items primarily • The scope of the system does not include a solution to the portability issue • The system will retrieve only the product on the shopping list. • The RSG technique is not convenient because it cannot work without human control
GroZi 2011 [9]	×	×	√	×	×	×	• The system assists the user in recognizing products using an Android application (portability) • Verbally announce to the user the name of the products	• The user needs to select the product manually from the tablet screen to announce the name of the product • The system recognizes only the products in the library

GroZi 2012 [10]	×	×	√	×	×	×	• The system enhanced by adding the feature of auto detection of the products in the grocery store • The auto detection feature aimed to free one of the user's hands to hold the grocery product while holding the tablet with the other hand • The system more portable than previous one, cleaner solution, less error, and user-friendly	• The authors did not repair the problem of recognizing only the products in the library
Wearable Wireless RFID System for Accessible Shopping Environments [11]	×	×	√	×	×	×	• Assist the user by a wearable system that declare the products information in a real-time scene • The material of the product does not influence the performance of the tag reader. • The delay time from scanning the product until announcing the product information was considered not long • The system is a reliable tool for navigation	• The RFID tags many modifications in the store • The store server may need the system to collect many data • The PDA devices are not suitable to use for the visually impaired • There is no product search according to the user requested product
BlindShopping [12]	√	√	√	×	×	√	• The system applies mobile techniques to assist the blind users in grocery shopping • The users claimed that the system was easy, and the verbal commands were useful • Pointing the camera to the QR codes was plausible for the users than pointing to the barcodes	• The system considered to be high cost and rough to deploy • the system will consume all the user physical power • There is no retrieval of products according to the user intentions

2.3 Object Recognition

The Dictionary of Computer Vision and Image Processing [13] defined Object Recognition as, {a general term for identifying the objects observed in an image. The process can also include computing the objects image or scene position, or labeling the image pixels or image features that belong to the object}. Furthermore, Object Recognition can be used to recognize and locate objects in images based on earlier information about the appearance of the objects [14].

General Object Recognition is split into two broad categories: Instance Recognition and Category (Class) Recognition. Instance Recognition comprises recognizing a 2D or 3D objects displayed from a new viewpoint, alongside a clustered scene, and with a limited object view. Category Recognition is recognizing instances of objects with the same familiar class such as animals or furniture [15]. Instance Recognition can be used to detect objects in cluttered scenes such as detecting grocery products in grocery stores. On the other hand, Category Recognition is usually used as image classification method such as classifying letters in an image using a bag of characters. Optical Character Recognition is one of the Category Recognition applications.

There are a number of algorithms developed for Instance Object Recognition. The latest algorithms rely on image feature matching methods. The features were detected from both the images in the real scenes and the objects images and stored in a database. Then, image feature matching techniques are be used to match the features of the scenes against the objects features database. When an appropriate number of match occur, a geometric transformation aligns the matched features together [16].

Through image features, the similarity between two images can be determined using image properties. Image features are the function that can measure the characteristics of an object. There are two types of features: low-level features and high-level features. Feature algorithms can directly extract low-level features from the image. On the other hand, high-level features rely on the extraction of low-level features. In any image, different features such as points, edges, lines, and corners can be detected [17].

2.3.1 Interest Point Matching (IPM)

One of the images features used for matching images is interest points. Interest point matches the geometry of the comparable images using a transformation function that inputs the coordination of the matching points in both images. The point features can be called: interest point, the key point, the corner point, and control point. We will use the term interest point [18].

The applied interest point matching algorithms have three main steps [15]:

1. Interest Point Detection: Detects any feature in the image that is unique and may resemble a feature in the corresponding image with changed image situations.
2. Interest Point Description: Describes the points near the detect interest point locations to be matched with other descriptors.
3. Interest Point Matching: Matches between the interest point descriptors using distance as shown in Figure (2.8).

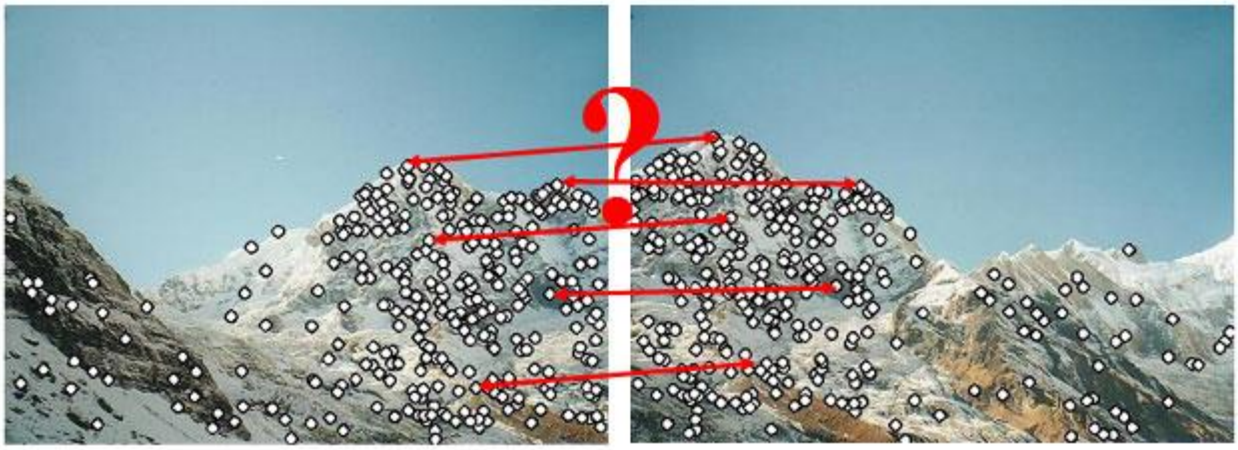


Figure 2.8: Interest Point Matching [15]

2.3.1.1 Interest Point Detectors

Over the years, the computer vision community created a number of interest point detectors. The most famous detector was invented in 1988 and was called Harries corner detector [19]. Harries was not scale invariant. Therefore, Lindeberg [20] in 1998 came to detect each interest point with its characteristic scale. He used Hessian and Laplacian matrix for blob detection (detect features pixels that are different from others). In 2001 [21], Mikolajczyk and Schmid created a robust and scale invariant interest point detector. They used Harris-Laplace and Hessian Laplace matrices. Lowe [22] focused on the speed of blob detection by calculating the Laplacian of Gaussian (LoG) by filtering the Difference of Gaussians (DoG).

By 2005 [23], Fast Hessian detector was introduced. Fast Hessian detector depends on the Hessian matrix. However, rather than using the Laplace detector for the scale measuring, it used Hessian for location and scale. The authors called it Fast Hessian detector because it relies on integral images to reduce the time of calculation. Figure (2.9) shows the detected interest points of Sunflower field using Hessian detector.

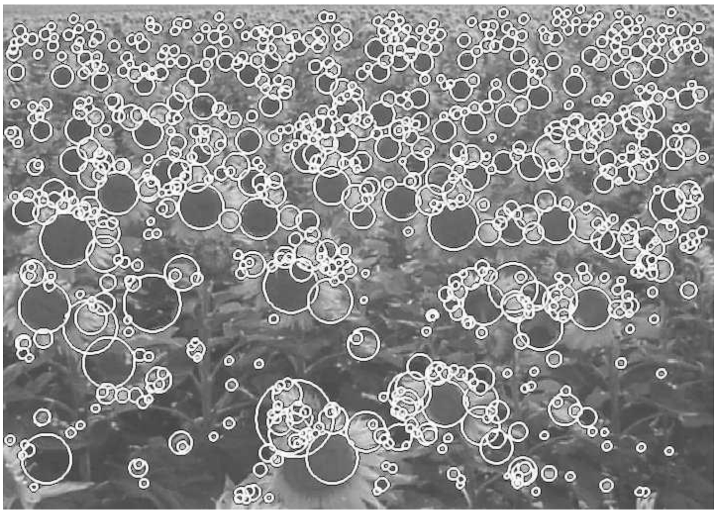


Figure 2.9: Hessian Detector Results on Sunflower Field [23]

Considering a point x=(x,y) available in image I, the Hessian matrix will be H(x,σ) in point x at scale σ as follows:

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \tag{2.1}$$

In the image I at the point x, $L_{xx}(x,\sigma)$ is the convolution of Gaussian derivation second order in x dimension and $L_{yy}(x,\sigma)$ is the convolution of Gaussian derivation second order in y dimension. In addition, $L_{xy}(x,\sigma)$ is the convolution of Gaussian derivation second order in xy dimension [24].

2.3.1.2 Interest Point Descriptors

As there are a number of interest point detectors, there are a number of interest point descriptors. The most known descriptor was Scale Invariant Feature Transform (SIFT). Lowe [25]introduced SIFT in 2004. SIFT aimed to describe the interest point neighbors that contained small-scale features for calculating the histogram of locally oriented gradients for the pixels nearby interest point. The SIFT was advanced by Ke and Sukthankar [26] to apply Principal Component Analysis (PCA) on the gradient image and called PCA-SIFT. Mikolajczyk et al. [27] verified that PCA-SIFT is slower than

SIFT because of its slower feature detection. Gradient Location Orientation Histogram (GLOH) is an improved version of SIFT introduced by [27], but it was considered computationally expensive.

Bay et al. developed Speeded up Robust Features (SURF) descriptor [23] that uses the Fast Hessian detector. SURF depends on describing the Haar-Wavelet responses of the interest point neighbors. SURF consist of two steps. Based on the circular region around the interest point, the retrieved information is used to repair the orientation of the image. Then, a square region is created to be aligned with the retrieved orientation, and to extract the SURF descriptor from the result. The descriptor task starts by splitting the regions, usually in 4×4 square sub-regions. Regularly, at 5×5 sample points, a simple feature is calculated for each sub-regions. The Haar-Wavelet response in the horizontal direction is called d_x, while the Haar-Wavelet response in the vertical direction called d_y.

The descriptor measures the responses d_x and d_y with Gaussian centered at the interest point. The first record in the feature vector will be the responses dx and dy summed up over each region. To insert facts about the polarity of the intensity variations, the sum of the absolute value of the responses $|d_x|$ and $|d_y|$ will be calculated. So, each sub-region will contain a descriptor vector consisting of four-dimensional values v= $(\sum d_x, \sum d_y, |\sum d_x|, |\sum d_y|)$. Figure (2.10) shows the descriptor of sub-regions with different intensity patterns. The left sub-region is homogenous, the middle sub-region represents different frequencies in x direction, and the right sub-region represents a gradually increase intensity in x direction.

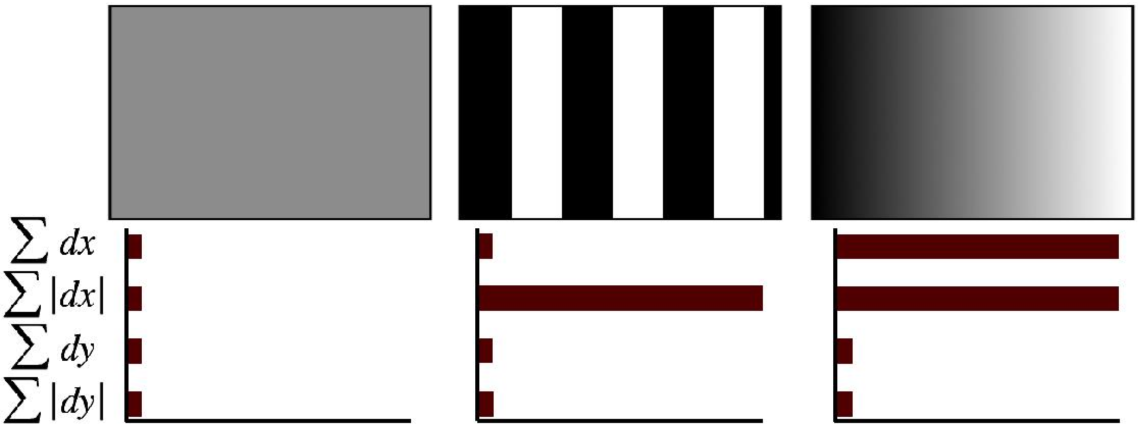


Figure 2.10: SURF descriptor vector values in a different intensity pattern sub-regions [23]

SURF was compared with SIFT, PCA-SIFT, and GLOH using Fast Hessian detector. The results showed that SURF performance in all the comparisons was better than the others. Based on the average recognition rate, SURF obtained 82.6%, GLOH obtained 78.3%, SIFT obtained 78.1%, and PCA-SIFT obtained 72.3%. SURF proved to be a fast and accurate interest point detector and descriptor. Figure (2.11) shows the SURF different scale rectangle descriptors of Graffiti scene.

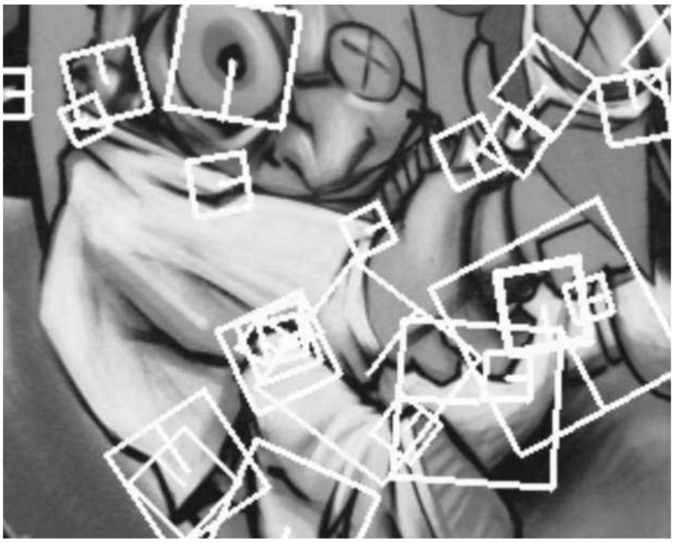


Figure 2.11: SURF descriptor of Graffiti scene [23]

2.3.1.3 Matching Strategies

There are three matching strategies that can be used to match the descriptors between two regions A and B [27].

1. Threshold Based Matching: The descriptors are matched if the distance between them is below the threshold.
2. Nearest Neighbor based Matching (NN): The descriptors are matched if the descriptor of region B is the nearest neighbor to the descriptor of region A. Also, the distance between both the descriptors is below the threshold.
3. Nearest Neighbor Distance Ratio Matching (NNDR): It is similar to NN, but the threshold is applied to the distance ratio between the first and second nearest neighbor. The regions A and B will be matched if: $\|D_A - D_B\|/\|D_B - D_C\| < t$ Such that D_A is the descriptor in region A. D_B is the first nearest neighbor to D_A in region B. D_c is the second nearest neighbor to D_A in region B. In all strategies the matching will be done on each descriptor of the reference image with each descriptor on the transformed image.

2.3.1.4 Interest Point Matching Applications

i) Recognizing Groceries in situ Using in vitro Training Data

The work in [28] contributes a new multimedia dataset called GroZi-120 containing 120 grocery products. Each product is recorded from two different situations: *in situ* images pull out from grocery store filming video clips, and *in vitro* images was collected from the web. The work applied object recognition and detection algorithms in GroZi-120 dataset. Then, the authors measured the localization rates for the obtained results.

The GroZi-120 database was created under different situations to measure the difference in quality between the training data and the testing data for the mission of object detection and recognition. The products in the database fall under two representations. The *in vitro* representation (Figure (2.12)) aims to capture images that are isolated and fall under ideal conditions such as stock photography studios or labs. The *in vitro* (Figure (2.13)) images was obtained from web. Then, the authors set the images background to transparency. The *in situ* images represent the images in the real world. Therefore, a video shot was obtained from a grocery store containing the same products in the *in vitro* images. Then, the video shot was extracted to image frames. The *in vitro* images was compared with the *in situ* images using object detecting and recognition algorithms.

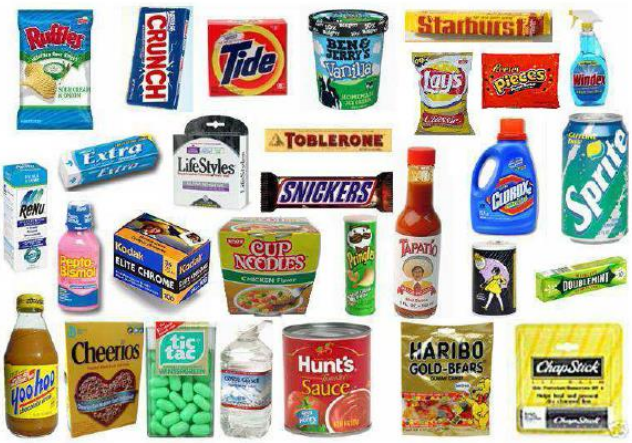


Figure 2.12: Example of in vitro images [28]

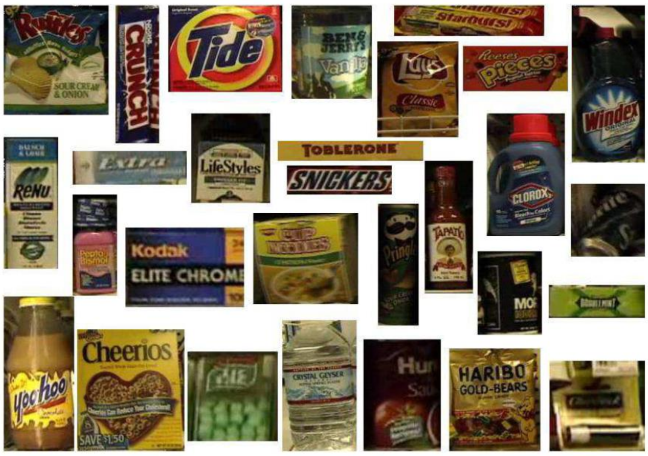


Figure 2.13: Example of in situ images [28]

There are number of algorithms used for object detection and recognition in [27] such as SIFT (mentioned in 2.3.1.2). In the SIFT algorithm, they compared the features of the *in situ* images with the features of the *in vitro* images. Then, evaluate the results using the localization rates (precision and recall). The SIFT performed better than the other algorithms due to the uniqueness of the products box text and symbols.

ii) Toward Real-Time Grocery Detection for the Visually Impaired (ShelfScanner)

Dr. Serge Belongie created a system for object detection called ShelfScanner [29] as an extension for [28]. ShelfScanner was used to aid the visually impaired people to shop independently without any person assistance. ShelfScanner relied on detecting the user required items recorded in an online shopping list. The detection was made by video streams retrieved from the camera. The user items were detected using object detection algorithm. The system used GroZi-120 dataset in experiments to detect products in shopping list containing ten items.

The system's software operated using a Mosaic algorithm based on Lucas-Kanade optical flow method to form a single image from a sequence of video frames. The system used an object detection algorithm for detecting the user requested items within the video frames. Speeded up Robust Features (SURF) descriptor (mentioned in 2.3.1.2) was used by matching the test images key points with the training key points. Then, homography is applied to the matching points in order to find the location of the product in the image. The hardware of the system consists of a camera carried by the user to scan the aisle shelves, and a powerful laptop transported in a backpack like system.

The inputs are represented by images of the shopping list products available in GroZi-120 dataset vitro images and the video frames from the user camera. The user passes the camera over the aisle shelves. The output is the number of points in the frames of the video that match the products in the shopping list.

The system was evaluated using a strict threshold. The results of the recognition rates from 52 item classes are: 17 easy items resulted with zero false positives, eight moderate items resulted between one and 100 false positives, and the last hard items resulted in more than 100 false positives. The authors concluded that, the system is truly usable according to the detection results. Although, an up to date training results could improve the recognition. The system has the advantage of using the mosaic technique in order to avoid handling a video scene multiple times. The authors recommend integrating with text detection to reduce the titubation of the product packaging [29].

ShelfScanner is restricted to find only the products on the shopping list. The system is limited only to detect objects on the shelves, not to detect aisles category. The user must pass the camera over all the aisle shelves, and that may stress the user's hands.

2.3.2 Optical Character Recognition (OCR)

Optical Character Recognition (OCR) is a technique used to detect and recognize texts in document images or scene images. There are number of engines developed for OCR.

The scanned text images are different from the scene text images according to the differences in font size, color, orientation, and the background disruption. K. Wang and S. Belongie [30] compared the performance of the two most known OCR engine (Tesseract and ABBYY Fine Reader) on ICDAR 2003 robust reading dataset and Street View Text dataset as shown in figure (2.14). The Street View Text dataset is outdoor image text dataset which shows high inconsistency and usually has low resolution.

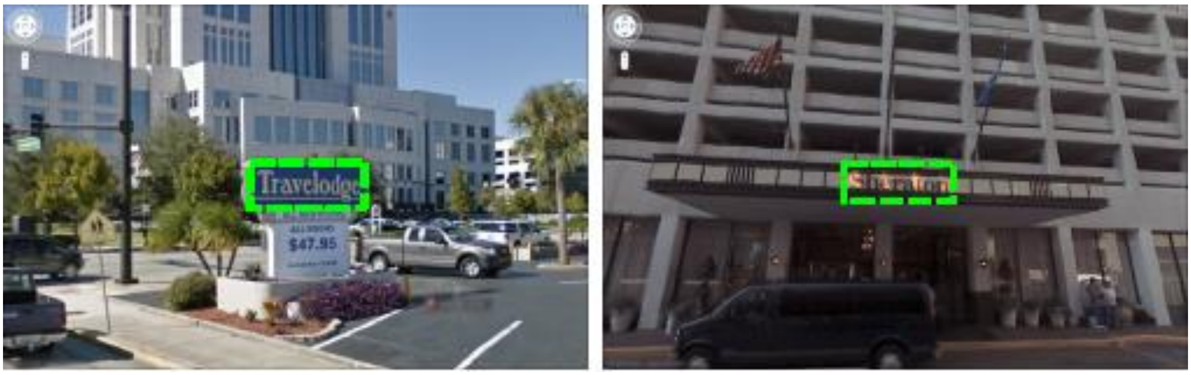


Figure 2.14: Example of Street View Text dataset [30]

The results of the test showed that the accuracy of the Tesseract was 31.5 % while the ABBYY was 47.7% [30].

2.3.2.1 ABBYY Fine Reader

The ABBYY Fine Reader [31] follows the following processing steps for character recognition in text images: i) intelligent background filtering; ii) adaptive binarization; iii) image resolution detection; iv) multilevel document analysis; v) character classifier and vi) structure differentiating classifier.

i) Intelligent background filtering

In this step, the text strings are separated from the background. The filter selects an optimal binarization parameter for each region. Figure (2.15) shows an example of background filtering.

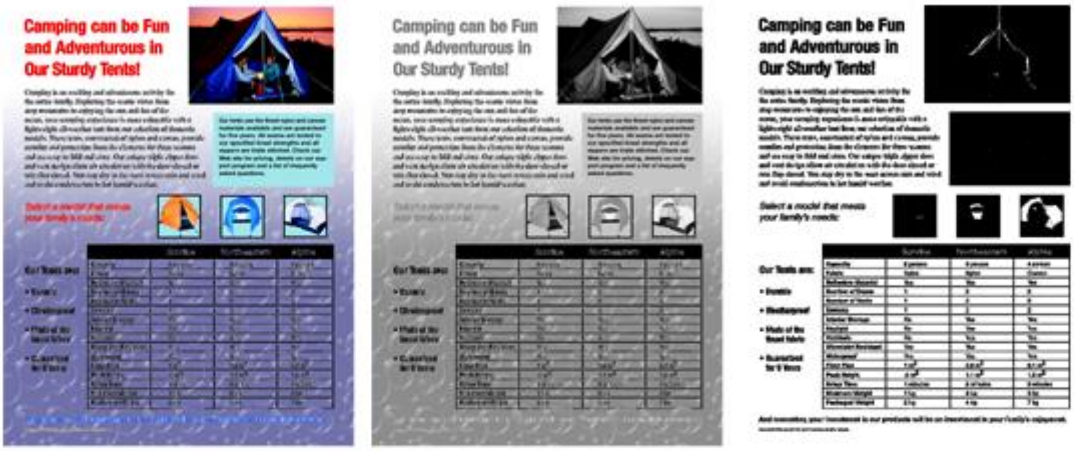

Camping can be Fun and Adventurous in Our Sturdy Tents!
Camping can be Fun and Adventurous in Our Sturdy Tents!
Camping can be Fun and Adventurous in Our Sturdy Tents!

Figure 2.15: Example of Intelligent Background Filtering [31]

ii) Adaptive binarization

Due to the low contrast of the images, text recognition quality may be affected. Measuring the background brightness and black ranges saturations one of the approaches that can solve this problem. In this step, the binarization factors are computed for each line's section. In other words, the binarization accurately detects the lines and words. Figure (2.16) shows the difference between an accurate and a non-accurate binarization process.

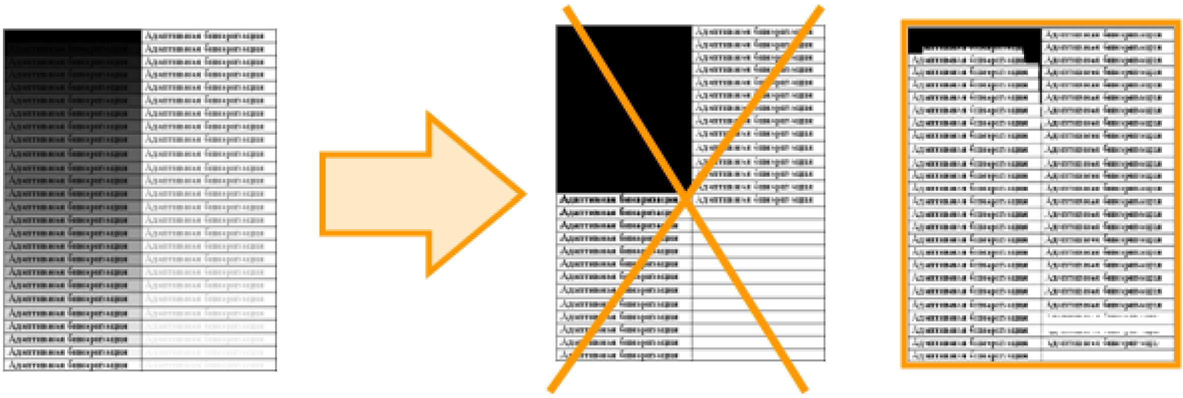


Figure 2.16: Adaptive Binarization in work [31]

iii) Image resolution detection

One of the reasons for low text recognition is low image resolution. This aspect can also lead to a slow recognition. If the user scanned a document or image, the image resolution would be set by Dot per Inch (DPI). On the other hand, if the document is digitalized, the Metadata containing the image resolution came with the image.

iv) Multilevel document analysis (MDA)

These days, the documents do not include only text but also images, tables, footer, and header. The OCR program usually starts by analyzing the structure of the document. Then, it reads the document. The structure of the document (represented in a hierarchy) contains:

- page
- table, text block
- table cell
- paragraph, picture
- line
- word, picture within a line
- Letter (character).

In the hierarchy, each object contains a smaller objects as shown in figure (2.17) such that the line is composed of number of words and words composed of a number of letters. The program starts by analyzing from top to bottom until it reaches the smallest object (character). Once the program reaches the smallest object, it reverses the task by starting from the lowest to the highest object. For that reason, the task called Multilevel Document Analysis.

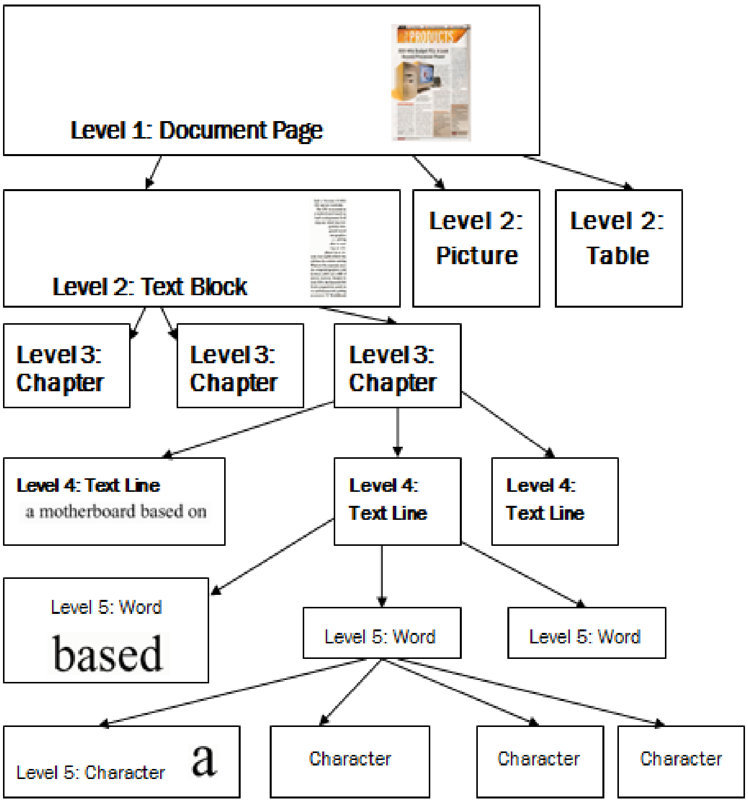


Figure 2.17: MDA Hierarchy [31]

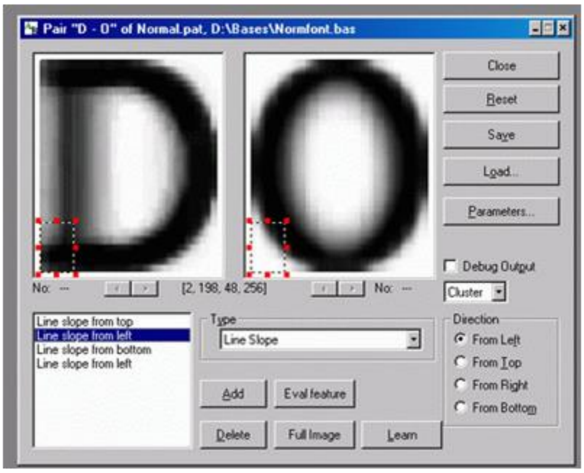
v) Character classifier

After dividing the words into characters, the character images are sent to be recognized by the classifier. ABBYY uses the following types of classifiers.

- *Raster classifier*: it compares the letter image with a set of pattern images. The pattern contains number of writing ways of each character. ABBYY decides which character matches if a matching letter was found in the pattern. The Raster classifier works fast, but it does not give very accurate results. Most of the OCR programs uses the Poster classifier.
- *Feature classifier:* it resembles the raster classifier in matching the letter image with a pattern letter image. However, it depends on extracting letter features such as perimeter, black dots, number in a specific area or specific line. OCR programs frequently use it. The

classifier is considered as fast as the Raster classifier. The accuracy of the classifier depends on the selected features of the letter.

- *Contour classifier*. The contour classifier is a part of feature classifier, but it detects the features on the letter contour. It is considered fast for recognizing decorative fonts such as Gothic script.
- *Structure classifier*. The Structure classifier is developed by ABBYY to recognize handwritten letters. It decomposes the letters into a number of components such as lines, arcs, circles, and dots. Then, it reconstructs the letter using the decomposed components. Finally, the reconstructed letter is compared with a set of pattern structures. The classifier is considered slow comparing with Raster and Feature classifiers, but it is more accurate. Its advantage is that it can construct the missing parts from letters.
- *Feature differentiating classifier*. The classifier is used to distinguish between similar letters but not to classify the entire image. As shown in figure (2.18), the classifier computes the feature parameters to indicate the difference between the two letters. It computes the slope of the lines in the letters.



Pair "D - 0" of Normal.pat, D:\Bases\Normfont.bas

No: --- [2, 198, 48, 256] No: ---

Line slope from top
Line slope from left
Line slope from bottom
Line slope from left

Type: Line Slope

□ Debug Output

Cluster

Direction
● From Left
○ From Top
○ From Right
○ From Bottom

Figure 2.18: Example of Feature Differentiating Classifier [31]

- *Structure differentiating classifier*. The classifier is considered very accurate, and it used to detect hand written text. It is also used to distinguish between very similar letters. The classifier is more accurate because it detects the structure of the letters.

2.3.2.2 Tesseract OCR Engine

The Tesseract OCR engine is an open source HP research developed between 1984 and 1994. The Tesseract appeared in the UNLV annual test of OCR accuracy conducted in 1995 [32]. HP released Tesseract as open source in 2005.

The Tesseract architecture [33] shown in Figure (2.19) staged as i) Adaptive Thresholding; ii) Connected Component Analysis; iii) Find Text Lines and Words; and iv) Recognize Word Pass one; v) Recognize Word Pass two [33].

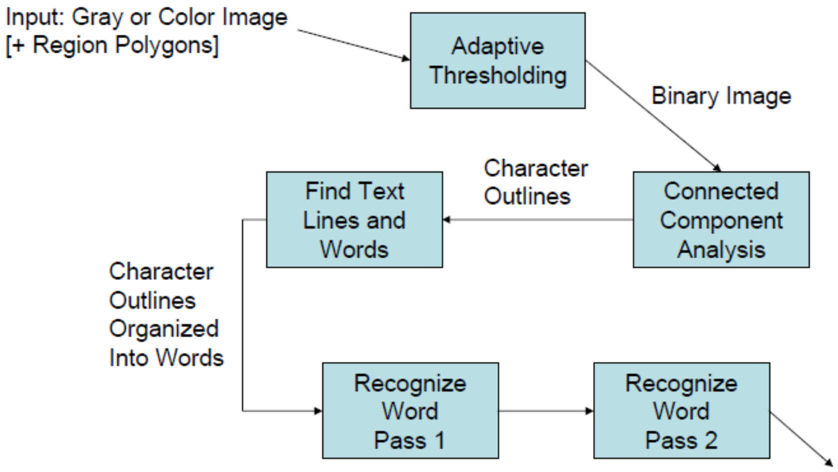

Input: Gray or Color Image
[+ Region Polygons]
Adaptive Thresholding
Binary Image
Connected Component Analysis
Character Outlines
Find Text Lines and Words
Character Outlines Organized Into Words
Recognize Word Pass 1
Recognize Word Pass 2

Figure 2.19: Tesseract Architecture [34]

vi) Adaptive Thresholding

The Adaptive Threshold aims to convert a gray or colored image to binary image.

vii) Connected Component Analysis

The Connected Component Analysis stage aim to store the outlines of the components. It is useful in case of nesting outlines, and number of child and grandchild outlines. The outlines are grouped in nests and stored in blobs.

viii) Find Text Lines and Words

- *Line Finding*. The process main part is blob filtering and line construction. First, the system will filter the blobs according to the page layout analysis. Second, the filtered blobs allocated to a specific text line. After that, the baselines was calculated to fit back the blobs into the right line.
- *Baseline Fitting*. The stage fit the baselines more accurately by applying quadratic spline on the text lines. The stage helps in handling pages with curved baselines.

- *Fixed Pitch Detection and Chopping*. Test the text lines if they are fixed bitch. Then, using the bitch, it chops the words to characters. Finally, send the words for word recognition.
- *Proportional Word Finding.* If the pitch was not fixed, that mean the gap between the words is too small, and the system will not be able to separate between words. The solution is to calculate the gap between the vertical range between the baseline and the mean line. If the space was close to the threshold, the decision would be taken after word recognition.

ix) Recognize Word Pass one

The architecture of recognizing words is shown in Figure (2.20).The word recognition phase aim to show how the word segmented to characters. The stages of word recognition are applied to non-fixed pitch text. The phase contains the following stages:

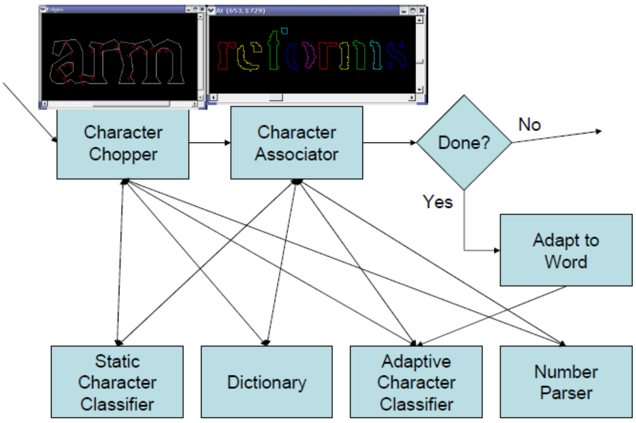


Figure 2.20: Word Recognition Architecture in Tesseract [34]

- *Chopping Joined Characters*

Tesseract chops the blob with the worst confidence in order to improve the results. The candidate chop points are calculated by polygonal approximation of the outline using concave vertices. Then, to effectively distinct joined characters it may require up to three pairs of chop points.

- *Associating Broken Characters*

If the word is still not good after chopping, the associator will be introduced. The associator will search for the best first candidate character according to the chopped blobs.

- *Static Character Classifier*

The classification is done using two steps. First, a short list of the character classes that may match the unknown character was created called class pruner. Then, the features similarity between the class pruner and the unknown character was calculated.

- *Linguistic Analysis*

The linguistic module was considered whenever a new word recognition module appeared. The linguistic module selects the best word string fall under the following categories: Top frequent word, Top dictionary word (Dictionary), Top numeric word (Number Parser), Top upper case word, Top lower case word, Top classifier choice word.

- *Adaptive Classifier*

The adaptive came to fill the gap of the static classifier weakness. The static classifier is good at font type generalizing. Although, the static classifier is a week in discriminating between different characters and between character and non-character. The adaptive classifier is able to discriminate in any document according to the input from the static classifier.

x) Recognize Word Pass two

In case of words did not recognize well, a second pass will be run to recognize the unknown words.

2.3.2.3 OCR Engines Applications

i) End to End Scene Text Recognition

The work [35] introduced the evaluation of two systems. First, a system was conducted from the text detection stage followed by OCR engine recognition. The text detection phase used state of art text detector called Stroke Width Transformation (SWT). The used OCR engine is the ABBYY OCR engine (mentioned in 2.3.2.1). Second, an extension system was introduced according to an earlier system. The advanced system is a word detector based on lexicons and training data. The results show that the performance of the second system is better than the first one. Although, the authors stated that SWT and OCR system gave better results in text detection such that they read 438 words correctly from 482 words.

ii) Video Text Detection and Recognition: Dataset and Benchmark

The system emphasis on text detection and recognition in video. They extend the work of [35] to be evaluated along with the ABBYY OCR engine (mentioned in 2.3.2.1) and other text detection and recognition algorithms. The dataset used is ICDAR 2013 robust reading challenge three video dataset and YouTube Video Text (YVT) dataset. The evaluation for ABBYY was obtainable on the ICDAR dataset but not on YVT. The results of the Average Tracking Accuracy (ATA) of the ABBYY on the ICDAR dataset was minimal according to the other algorithms because ABBYY does not produce on lexicon like the other algorithms.

2.4 Performance Evaluation

There are numerous ways to measure the performance of any matching algorithms. One can be by counting the number of true and false matches using the confusion matrix values as shown in Table (2.3).

Table 2.3: Confusion Matrix [15]

	True matches	True nonmatches
System predicated match	True Positive (TP)	False Positive (FP)
System predicated non-match	False Negative (FN)	True Negative (TN)

In Table (2.3), the term TP stands for true positive values that mean the total of the system truly predicted matches. The term FP stands for false positive values that mean the total of the system falsely predicted matches. The term TN stands for true negative values that mean the total of the system truly predicted non-match. The term FN stands for false negative that mean the total of the system falsely predicted non-match.

From the confusion matrix values, the values of performance rates can be constructed. Recall, or True Positive Rate (TPR) indicates the total of the relevant documents retrieved. Precision or Positive Predicted Value (PPV) indicates the total of relevant retrieved documents. Fall out, or False Positive Rate (FPR) is the total number of falsely retrieved documents. Negative Predicted Values (NPV) indicates the total of irrelevant retrieved documents. Accuracy (ACC) means the total value of the system accuracy [15].

The equations of each performance rate as follows [15]:

$$\text{Recall } (TPR) = \frac{TP}{TP+FN} \tag{2.2}$$

$$Fall\ out\ (FPR) = \frac{FP}{FP+TN} \tag{2.3}$$

$$Precision\ (PPV) = \frac{TP}{TP+FP} \tag{2.4}$$

$$Negative\ Predicted\ Value\ (NPV) = \frac{TN}{TN+FN} \tag{2.5}$$

$$Accuracy\ (ACC) = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.6}$$

2.5 Multimodal Systems

A multimodal system was defined by [36] as; a multimodal system is a system that can use more than one source of data for recognition. The multimodal system can help in solving real-world recognition problems by combining multiple systems together. The main advantages of the multimodal systems are increased accuracy, fewer problems, and improved safety.

There are number of types of the multimodal systems such as i) multiple characteristics; ii) one characteristic with multiple sensors; iii) one sample with multiple algorithms; and iv) multiple impressions. In one sample with multiple algorithms, number of algorithms used to match one sample. The advantage of using multiple algorithms is that each algorithm can cover the weakness of the other ones.

Each type of multimodal systems can operate in two modes: serial mode or parallel mode. In serial mode, the first algorithm will be applied to the sample and if it fails, the next one is be applied. The final result is the result of all the algorithms used fused together. In parallel mode, every algorithm is applied at the same time [36].

Each recognition system, as shown in figure (2.21), reads a sample to extract the features and compare them with the template features in order to produce a matching score. The matching score describes the similarity between the sample and the template.

Then, the matching score is compared with a threshold to determine if the recognition process succeeded [37].

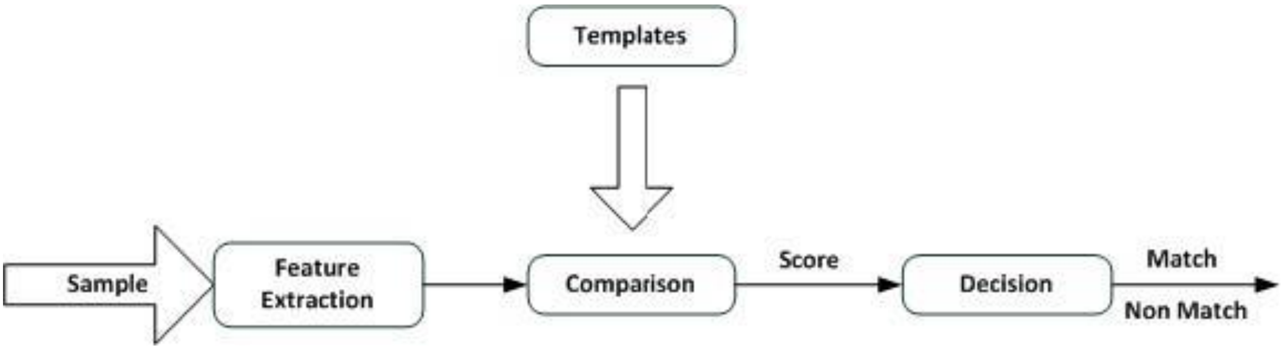


Figure 2.21: Single Recognition System [37]

The mission of combining different algorithms to construct a multimodal system is called fusion. There are number of levels for multimodal systems fusion described as follows [36] [37]:

2.5.1 Feature Level Fusion

In this case, each algorithm extracts a number of features which can be of different types.The level starts when each algorithm extracts the features vectors that contain a description of each feature. If the samples are of the same type, the feature fusion combines the feature vectors into a new reliable one. On the other hand, if the samples are from different types, the feature fusion will concatenate the feature vectors into a new and detailed feature vector. Figure (2.22) shows the stages of the feature level fusion.

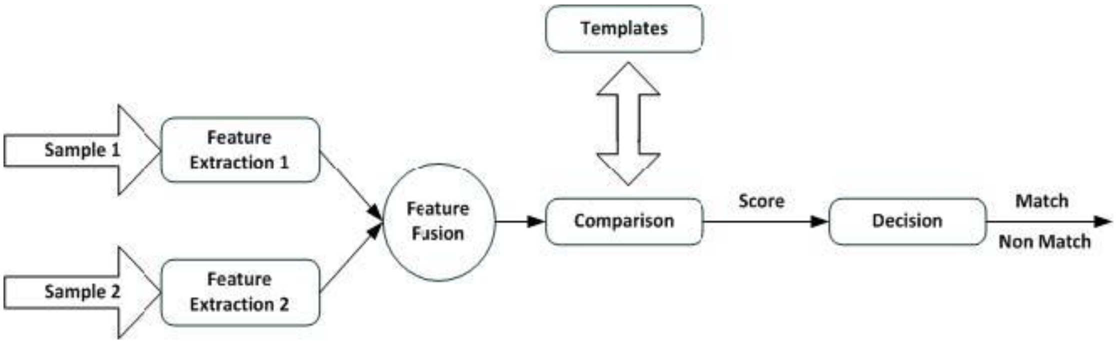


Figure 2.22: Feature level fusion [37]

2.5.2 Score Level Fusion

The main advantage of score fusion is that there is not needed to know any prior information about the features or features extraction methods. In the score fusion, the combination will proceed after gaining the similarity score. This combination step can be performed using two different techniques: classification or score combination.

The classification technique verifies the algorithms as a classification problem: either accepting or rejecting the similarity score. The classification feature vector is represented by the similarity scores of different algorithms.

The score combination technique combines the similarity scores gained from different algorithms and either taking the average or select the minimum or maximum value. The weighted average is considered the best method in order to its high performance and simplicity. Some of the algorithms do not produce scores from the same range; that is why algorithms apply scores normalization. Score normalization is the procedure to convert the scores from different algorithms to a standard distribution. Figure (2.23) shows the score level fusion technique.

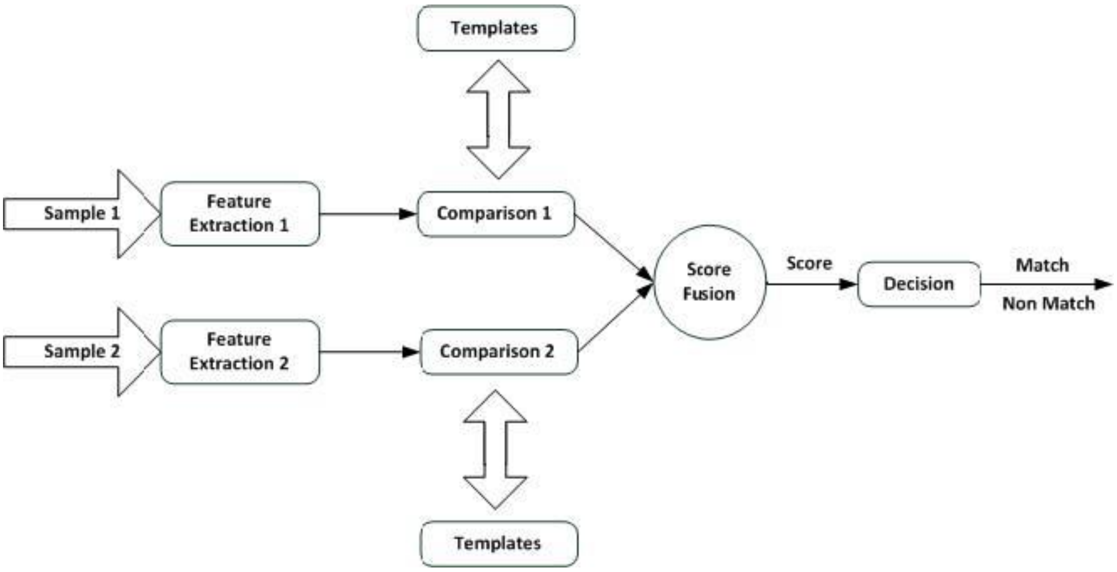


Figure 2.23: Score Level Fusion [37]

2.5.3 Decision Level Fusion

In the decision level fusion as shown in Figure (2.24), each system individually makes their decision about the similarity score either accept or reject. The decisions from different systems are then joined to one decision. Finally, the multimodal can either accept or reject the final decision.

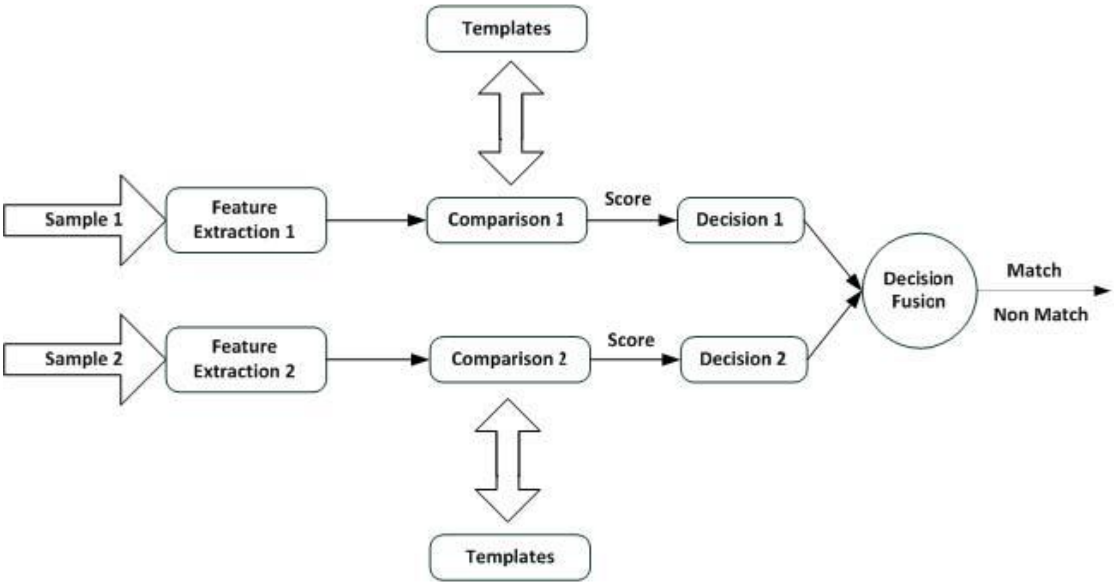


Figure 2.24: Decision level fusion [37]

There are number of methods that can be used in the decision fusion [38]: “AND” and “OR” rule, Majority Voting, Weighted Majority Voting, Bayesian Decision Vision, Dempster-Shafer Theory of Evidence, and Behavior Knowledge Space.

2.5.3.1 “AND” and “OR” Rule

The “AND” and “OR” method is considered the simplest method for decision fusion. If all the algorithms matches agreed on the similarity between the input sample and the template, then the “AND” rule will output a match found. While if at least one of the algorithms matches agreed on the similarity between the input sample and the template, then the “OR” rule outputs the match found. The “AND” rule usually leads to low False Accept Rate (FAR) and a high False Reject Rate (FRR). The “OR” rule usually leads to high FAR and a low FRR.

2.5.3.2 Majority Voting

The majority voting is the most common method used for decision fusion. The majority voting is applied if the majority of the algorithms matches agreed on the similarity between the input sample and the template. If there are R algorithm matches, the input sample assumes to match the template when at least k of the matches agreed on the similarity.

$$k = \begin{cases} \frac{R}{2} + 1 & if\ R\ is\ even \\ \frac{R+1}{2} & otherwise \end{cases} \tag{2.7}$$

In case no match is found from anyone from the matches, a reject answer is retrieved from the system. The advantage of this method is that no knowledge about the matches is, and no training is necessary to get a decision from the system.

2.5.3.3 Weighted Majority Voting

In case the algorithm matches are not from the same recognition accuracy or from different classifiers, the weighted majority voting is applied. The more accurate matches assign a higher weight to the other matches to equalize the accuracies. The matches output is converted into a degree of similarity based on M classes as follows.

$$s_{j,k} = \begin{cases} 1, & if\ output\ of\ the\ j^{th}\ matcher\ is\ class\ w_k \\ 0, & otherwise, \end{cases}$$

(2.8)

Such that, j=1,…..,R and k=1, …..,M. To compute the discriminating function for class w_k using the weighted voting, the following formula is used:

$$g_k = \sum_{j=1}^{R} w_j s_{j,k}, \tag{2.9}$$

Such that, for each j^{th} matches a weight is assigned to it as w_j.

2.5.4 Fusion Application

The work proposed to implement the text recognition along with the visual class recognition [39]. A new algorithm for text detection in the real scene was created. The new algorithm is based on saliency cues. The text recognition system compared the results of ABBYY OCR engine (mentioned in 2.3.2.1), Tesseract (mentioned in 2.3.2.2), and the output of the saliency cues as input to ABBYY. The text recognition stage was tested on ICDAR 2003 dataset. In the evaluation, the valued of the precision and recall was calculated. The results showed that the saliency method combined with ABBYY performed better than the other algorithms. The saliency system was evaluated on PASCAL VOC 2011 dataset. The new saliency method was compared to the state of art saliency methods. The results showed that the new saliency method performed better than the other methods. The multimodal system was implemented on IMET dataset. The multimodal system was experimented with the Bag of Words (BOW) model and SIFT features (mentioned in 2.3.1.2). The text recognition part in the multimodal was evaluated under two setups stages: i) input the result of the saliency model to OCR; ii) input the saliency result for the ground truth box (exact framing around object instance) to OCR. This method called OCR_bb. The accuracy results were measured under the following conditions: OCR, OCR_bb, BOW, BOW+OCR, and BOW+OCR_bb. The results showed that when using the BOW and OCR_bb together, the system performed better than the other systems.

2.6 Conclusion

There was number of earlier systems created to assist the visually impaired persons in grocery shopping. Some systems require hardware implementation either in the grocery store or as wearable devices. Therefore, these systems may require costly implementations or energy consuming for the users. Other systems require grocery products database construction or require wireless connections. Consequently, these systems may fail in recognizing products outside the database or fail under non-wireless connection areas. A comparison was made to measure the advantage and the disadvantages of each system.

In order to recognize objects in different scenes, there are number of object detection algorithms. We mentioned the two main algorithms in our thesis: IPM and OCR. Two of IPM algorithm was mentioned with their procedures and examples. Results of each example were introduced using the IPM. Additionally, OCR was described according to two famous OCR engines. Two research introduced to show the results of each OCR engines.

The multimodal system concept was explained to emphasize the concept of the fusion. Three types of fusion were explained with figures. We introduced the methods of computing the decision using equations. A work represents fusion concept between the visual features and text features. The results were compared with the different methods used for visual and text features.

Chapter 3

Methodology

Previous studies showed that there are number of limitations in the following systems: RoboCart [2] [3], Trinetra [6] [7], GroZi [8] [9] [10], Wearable Wireless RFID System for Accessible Shopping Environment [11], ShopTalk [4], ShopMobile [5], , Blind Shopping [12]. Some of them are not able to assist the users in selecting and retrieving a particular product from the grocery store shelves. Others do not retrieve the aisle category name and, therefore, the users may get lost the store. Some systems use different electronic devices to recognize the products or to navigate in the store such as barcode readers, RFID readers, or vibrating gloves. Some identify the products using data stored in a database that is hard to build or limited to specific products in the store. Various systems used the product identification codes such as the barcode that may not be placed under the products, or it is hard for the user to localize the code placement on the product. Others, exhaust the user with the different routines to reach their desired product such as asking the user to pass the camera manually over shelves by hands or passing a barcode reader over shelves.

The scope of this project is to build a device to assist the visually impaired in grocery shopping. The device consists of a shopping cart with three cameras installed on the cart. The system aims to inform the user with the name of the category he or she is located in and to advise him to locate his or her desired product on the shelf. The videos captured by the cameras on the cart represent the input data to the system. The system converts the video to a number of non-redundant images. The user announces his or her desired product name. Then, a speech to text tool is used to convert the speech to text. The system will compare two image recognition techniques: OCR(2.3.2) and IPM (2.3.1). The OCR technique recognizes the text written on the product labels to identify the user desired product. While, IPM compares the shelf view image with the product images. The method requires building a database of product images to be compared. The system searches the database for the product image based on the name of the user desired product. The system was implemented on the item master dataset. The dataset contains a number of product images captured from different views. The marketing view was used to create the environment resemble the shelf view image. While, the planogram front product images was used as the database required for the matching process in IPM.

The chapter is organized as follows: Section (3.1) introduces the system objectives. Section (3.2) describes the system environment. Section (3.3) explains the system design including the system workflow stages. Section (3.4) introduces the dataset that will be used in the testing phase. Section (3.5) describes the proposed analysis method for analyzing the results. Section (3.6) defines the fusion techniques according to our system results. Finally, Section (3.7) concludes the chapter with the main aspects and findings.

3.1 System Objective

In this research, our aim is to solve some issues that can disrupt the user while using the system. The proposed objective is to develop a software system that can achieve the following:

1. The system can work in any grocery store without any further implementation.
2. The visually impaired people can shop in the grocery store like any shopper using only the shopping cart.
3. The system announces to the user the name of the products category (grocery store section) in the aisle.
4. The system retrieves to the user desired product location in the aisle shelves.

The novel study is to compare two object recognition algorithms: OCR(2.3.2) and IPM (2.3.1) on grocery store products dataset. The comparison is aimed to justify which method is applicable for implementation and to show the result of both methods. A fusion multimodal was created as a novel study to fuse between the IPM and OCR. Furthermore, to state if the fusion multimodal will perform better results than each algorithm alone.

3.2 System Environment

The proposed system will be applied in grocery stores. Each grocery store is divided into aisles. Each aisle is organized according to a specific category (Ex. Soups, Herbs). Inside each aisle, the products are organized by brand. The system starts when the user enters any aisle as shown in Figure (3.1). It is designed not navigate the user from the grocery store entrance or to the cashier.

Figure 3.1: The system environment [40]

The user pushes a shopping cart consisting of three cameras installed vertically on the right side of the cart. This mechanism allows to scan all the aisle products from top to the down of the aisle. Each camera is named to indicate its position in the cart (Top, Middle, and Bottom) for later processing. The shopping cart is shown in Figure (3.2).

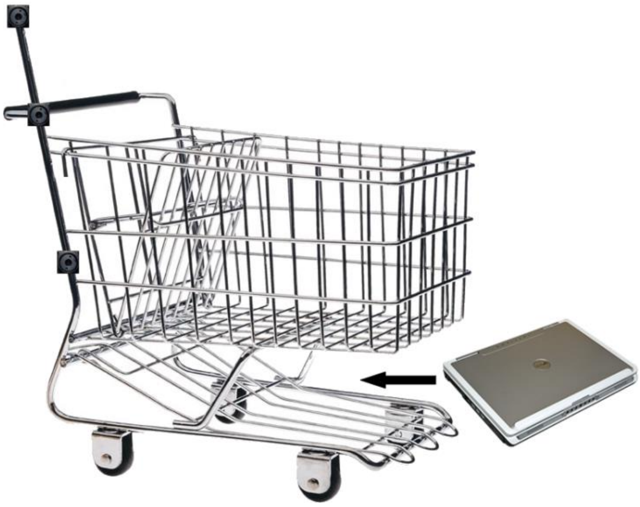


Figure 3.2: System Shopping Cart

3.3 Conceptual Design/Research Planning

The system is started by the input of the three camera views of the supermarket shelves from the cart cameras in the form of video clips.Each video clip is converted to a number of image frames which do not contain redundant frames. Each frame is connected with the name of the camera that captured the view either Top Camera, Middle Camera or Bottom Camera.

The system workflow is shown in Figure (3.3). When the user enters the aisle, he or she does not know the aisle category and, therefore, the first task for the system is to announce the name of the aisle category. After that, the user has the option of either proceeding to a new aisle or selecting a product from the current aisle. The second task for the system is to retrieve the user desired product from the shelves. This task consists of two parts. First, the system must find the user required product on the shelf by image recognition. Second, the location of the founded product on the shelf must be retrieved. The location can help the system to guide the user in front the shelf.

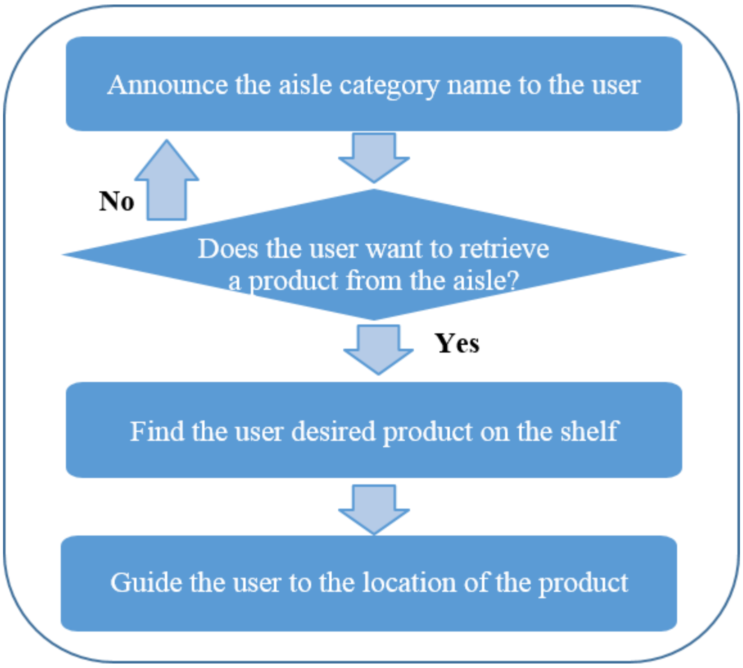

Announce the aisle category name to the user
No
Does the user want to retrieve a product from the aisle?
Yes
Find the user desired product on the shelf
Guide the user to the location of the product

Figure 3.3: System Workflow

3.3.1 Announcing the Aisle Category Name

While the user is walking in the grocery store, he or she needs to pay attention to the category name. The category name is beneficial because it links each of the desired product to purchase by the aisle category while he or she is walking through.

The task starts while the user is walking through a specific aisle, the three cameras in the shopping cart recording the view of the aisle shelves. The system inputs the cameras video clips to convert them to multiple frames and to eliminate the redundant ones.

The system performs OCR on the output image frames to extract the text from them. This text is supposed to include the text written on each product on the shelve view. The system compares the extracted text with the category name in the database (DB) which includes the common grocery stores category names and synonyms words for each category. For example, the category name Canned Vegetables and the synonyms Canned Corns or Canned Mushrooms.

The system announces to the user the most redundant category name. Afterward, the system asks the user either to continue the search for a new category or to retrieve a product from the current category. If the user chooses to search for a new category, the system repeats the same task again. However, if the user selects to retrieve a product from the present category, the next task starts. Figure (3.4) shows the procedure of announcing the aisle category name to the user.

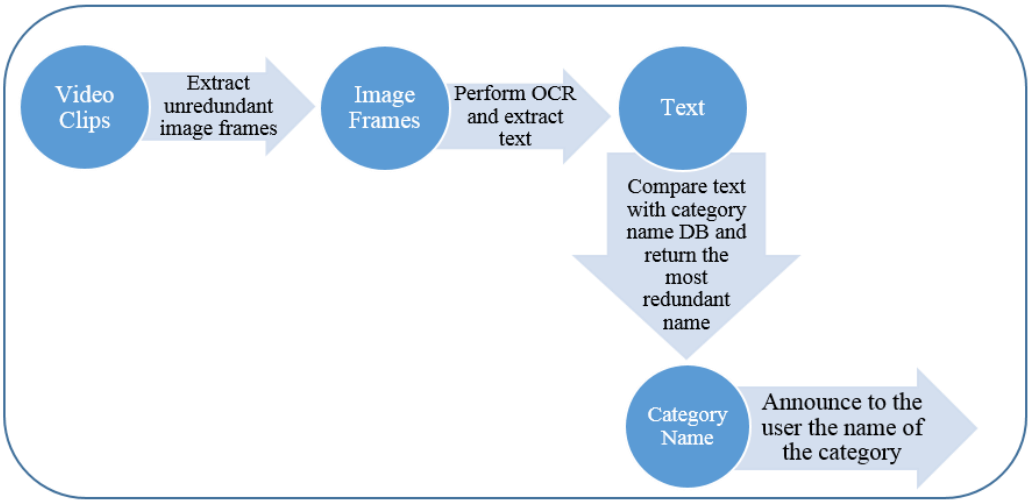

Video Clips
Extract unredundant image frames
Image Frames
Perform OCR and extract text
Text
Compare text with category name DB and return the most redundant name
Category Name
Announce to the user the name of the category

Figure 3.4: Define the category name to the user

3.3.2 Finding the Desired Product on the Shelf

This task starts after getting the acknowledgment from the user to search for a product within the present category. The system waits for the user to announce the name of the product. A speech to text tool is used to convert the speech to text. The name usually contains the product brand name, product category and sometimes additional information about the product. For example, Hershey's Syrup Chocolate Flavor, Hershey's is the product brand name, Syrup is the product category, and Chocolate Flavor is the additional information about the product.

The system retrieves the image frames of the shelves for the whole category. In order to find the user desired product in the image frames using the product name, we needed to use image recognition algorithms. We employed two image recognition algorithms: IPM (for more details please see chapter 2.3.1) and OCR (for more details please see chapter 2.3.2) on product images. We compare the result of both algorithms. In both cases, the system retrieves the location of the founded product in the image view in order to guide the user to the product location. This process is shown in Figure (3.5).

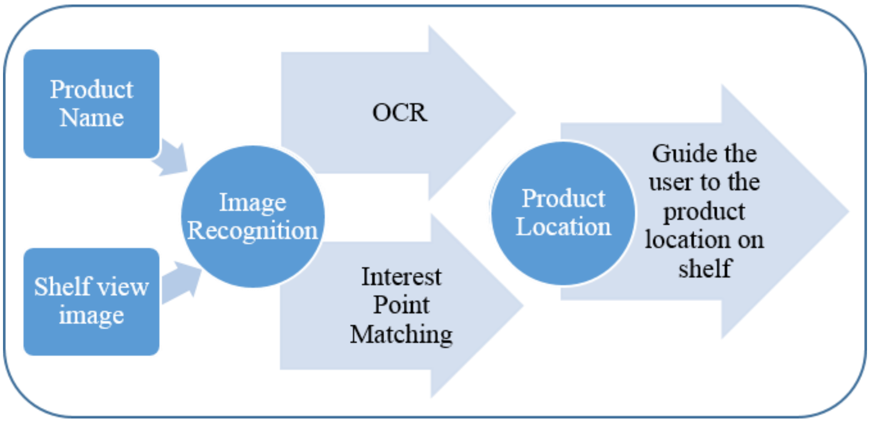


Figure 3.5: Find user desired product on shelf

3.3.2.1 Object Detection using Interest Point Matching (IPM)

Object detection is a method used to recognize objects within an image. In our case, the objects are the grocery store products. The aim of this task is to help the user to find a specific product within an image retrieved from a camera view.

As discussed in the literature section (2.3.1), there are different methods to detect objects within an image. One of these methods is to detect image features and extract feature descriptors. There are a number of features in any image such as edge, corner, and point. We detect interest points and extract feature descriptors at the interest points on the shelf view image and the products images. We build a database of products images in order to match it with the products on the grocery store shelves.

In the pre-processing stage, the scope is to detect the interest points and extract feature descriptors at the interest points of all the product images in the database as shown in Figure (3.6). The feature descriptor provides the feature vector and the feature location. The system stores both values in the product images database with the relevant image record.

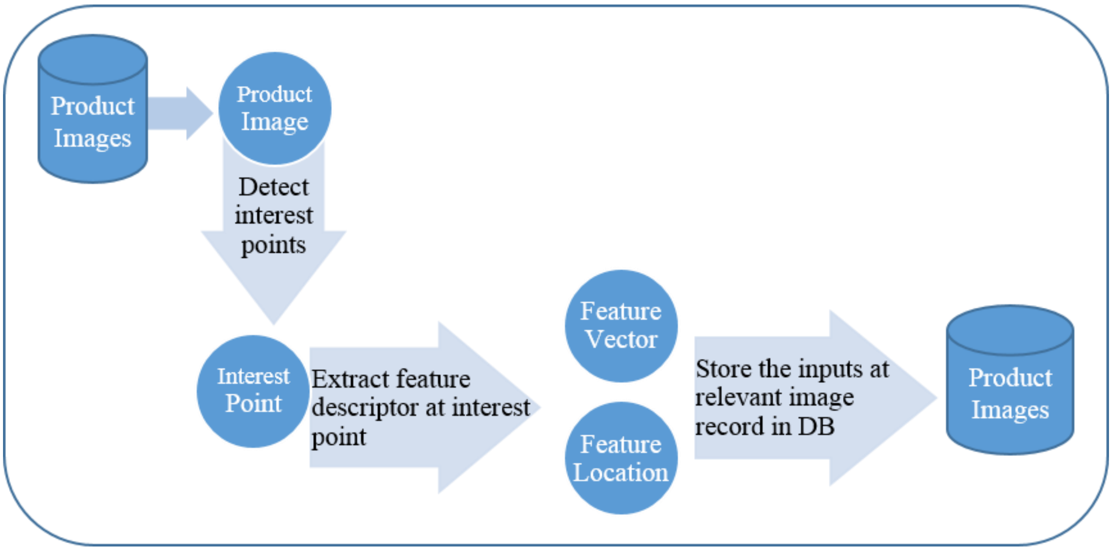


Figure 3.6: Preprocessing stage

The system starts by waiting for the user to announce the name of the product. The speech is then converted into text using speech to the text tool. The system searches for the desired product by name in the product images database. Then, retrieves the desired product image and feature vector and location for later processing.

When the system gets a shelf view image from the camera, the image is processed to detect the interest point and to extract the feature descriptor at the interest point. Based on the feature descriptor the feature vector and feature location are calculated. After that, retrieved product image and the shelf view image is matched using the feature vector of both images. The system returns the feature locations of the feature vectors that matched together. The matching locations in the product image are connected with the matching locations in the shelf image, and the outlier (the values that are distinctly separated from the rest matching locations) is eliminated.

Finally, the system draws a polygon around the matching location of the product on the shelf image to indicate the location of the product on the shelf. If the system did not find a match, a new shelf view image is retrieved and the process is repeated until the user enters a new category area. The general workflow of this process is shown in Figure (3.7) while Figures (3.8) and (3.9) detail some examples.

In order to detect the interest points, we used the Hessian Detector and the SURF descriptors, as detailed in section 2.3.1 of the current manuscript. To match between the descriptors, we use two matching strategies (2.3.1.3): NN and NNDR. The threshold scale is from zero to one. We experimented 5o testing samples under different threshold values. When assigning the threshold too high, it will result in a many false positive and if we assigned the threshold too low, it will result in many false negatives. The Computer Vision Algorithms and Applications book [15] agreed on the same point of view. Therefore, We assigned the threshold to (0.6) because it gave us better true matching results. The matching was done based on matching interest point in objects and not per interest point.

The IPM system was implemented using a MATLAB code performing the earlier described IPM procedure.

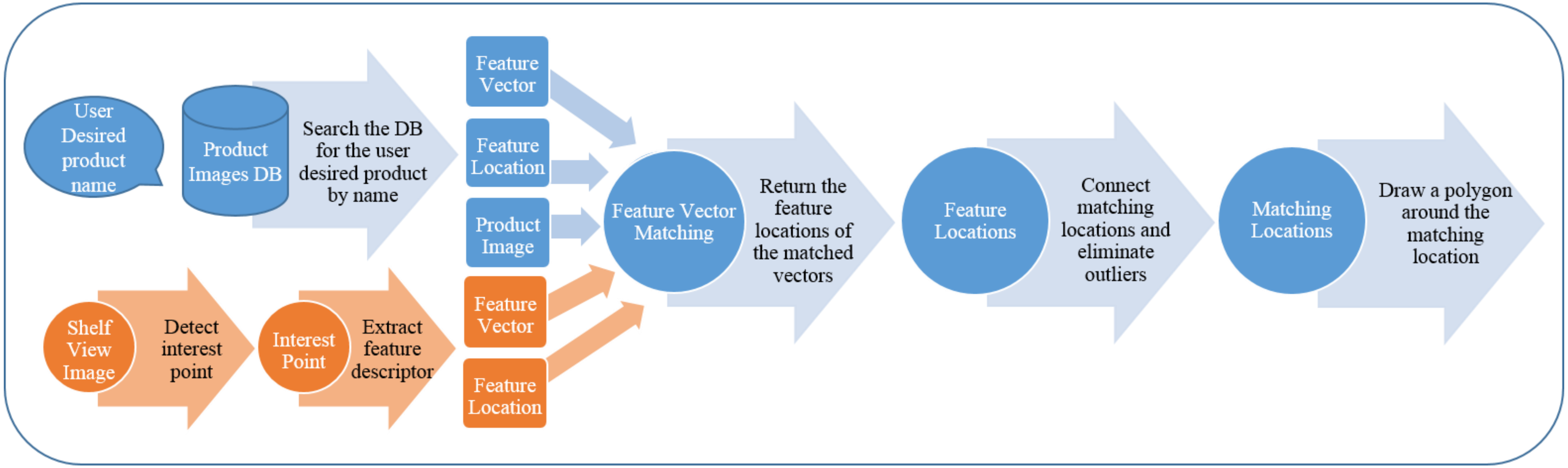

User Desired product name
Product Images DB
Search the DB for the user desired product by name
Feature Vector
Feature Location
Product Image
Feature Vector
Feature Location
Shelf View Image
Detect interest point
Interest Point
Extract feature descriptor
Feature Vector Matching
Return the feature locations of the matched vectors
Feature Locations
Connect matching locations and eliminate outliers
Matching Locations
Draw a polygon around the matching location

Figure 3.7: Object Detection using IPM workflow

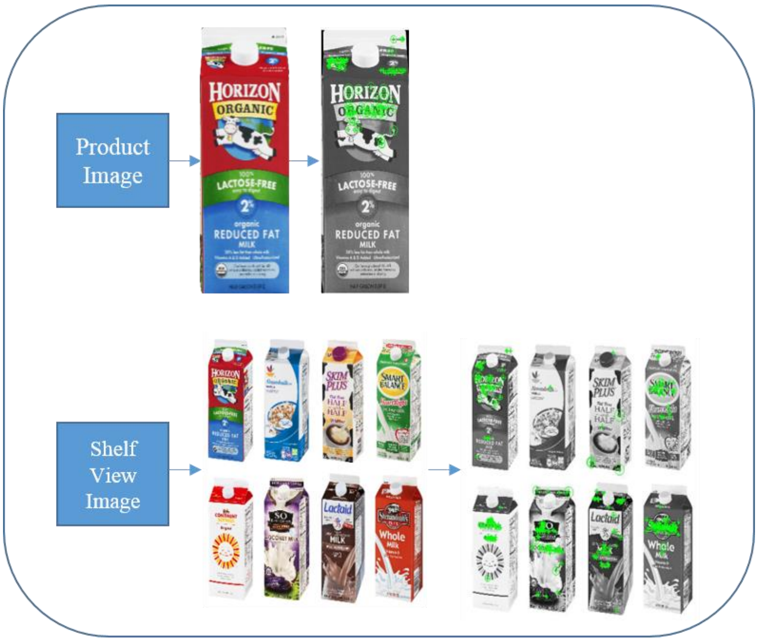


Figure 3.8: (Top) preprocessing stage (Bottom) shelf view image feature detection and extraction

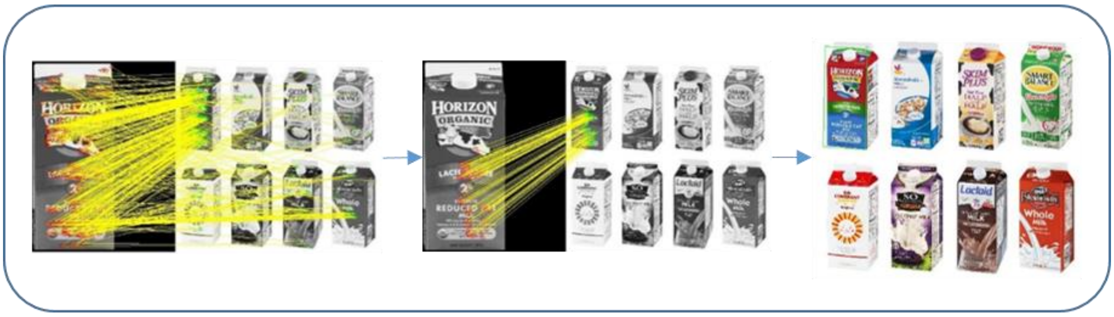


Figure 3.9: IPM example

3.3.2.2 Optical Character Recognition (OCR)

The optical character matching phase aims to detect and recognize the text written on the products labels while they are on the shelf. In this phase, we do not need to build a database of the product images, but instead use an OCR approach.

First, the system receives the shelf view image from the camera and the name of the desired product from the user. A speech to text tool is used to convert the speech to text. Then, the system extracts each product image from the shelf image view separately, to store it in its corresponding location in the image. The extraction is performed in two forms: the marketing form (MK) of the products and the front face of products (PF). Each product image in the two forms is sent to the OCR engine in order to recognize the text written on the product images. Then, the system stores each text with its relevant image. After that, the product name of the desired product is matched with the retrieved text. The matching is done twice, with text retrieved from the marketing image and with the text retrieved from the front face image. Furthermore, in each matching phase, the system matches the brand name together with the product description, and with the product description alone. If a match is found, the location of the product in the image is retrieved. If a match is not found, a new shelf view image is retrieved and the process is repeated until the user enters a new category area.

The process is shown in Figure (3.10). In order to perform text recognition, ABBYY Fine Reader OCR engine (2.3.2.1) was used. The OCR engine was plugged in with Eclipse using Java code. We first created an account in the OCR engine and registered in an application. Then, we get an ID and password to use later as credentials information. The credentials were the input gate to OCR engine. We created a Graphical User Interface (GUI) to insert the credentials file and upload the product images file to send to the OCR engine and retrieve the results as a text file.

When matching the product name given by user input with the retrieved OCR results from the engine, we used a threshold (0.6) similar to the IPM threshold. We will match the user desired product name under two circumstances: match the product Brand Name (BN) along with the Product Description (PD), and match the PD alone. For clarification, if 60% of the product description text exists in the OCR results it was accepted as match in case of product description matching. If we matched the brand name and product description with the text retrieved from the OCR, the full brand name must be retrieved from OCR and the product description must apply under the threshold rule. A total of 1550 images was send to the ABBYY OCR engine using an interface. The interface was built using Java and a log in credentials to enter the ABBYY engine.

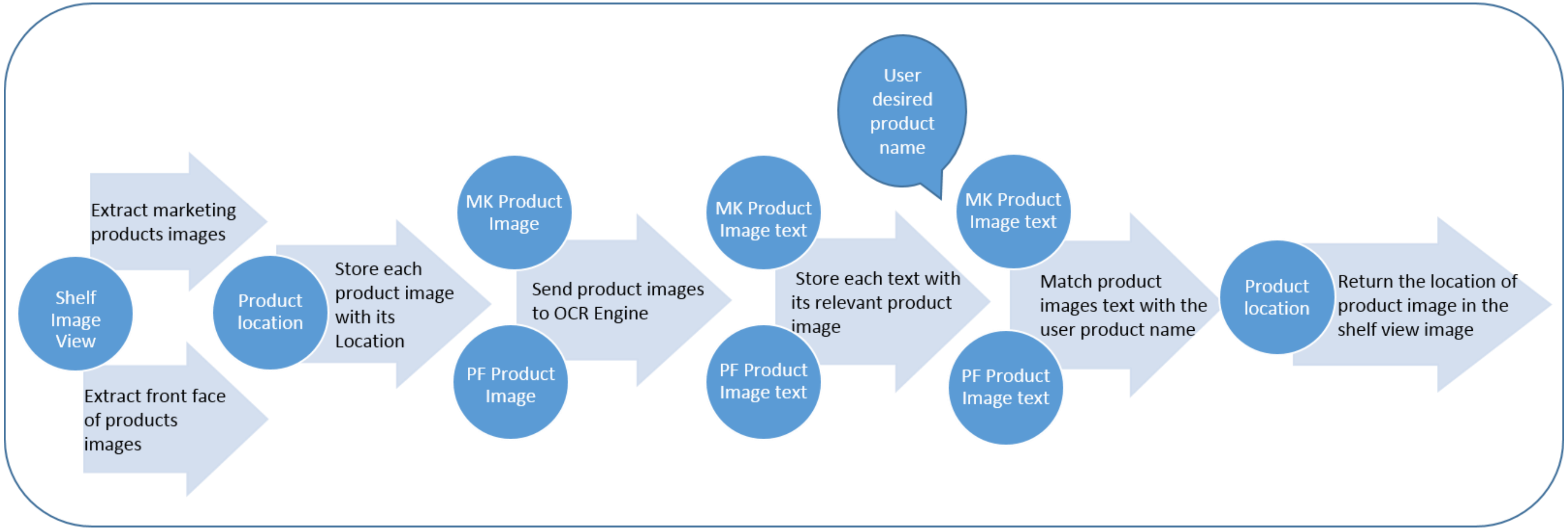

User desired product name
Extract marketing products images
Shelf Image View
Extract front face of products images
Product location
Store each product image with its Location
MK Product Image
PF Product Image
Send product images to OCR Engine
MK Product Image text
PF Product Image text
Store each text with its relevant product image
MK Product Image text
PF Product Image text
Match product images text with the user product name
Product location
Return the location of product image in the shelf view image

Figure 3.10: OCR procedure

3.3.3 Guide the User to the Product Location

In this stage as shown in Figure (3.11), the system will retrieve the camera name that captured the shelf view image that contains the founded product. Also, the system will retrieve from the previous stage the location of the product in the shelf image.

The task will start by asking the user to move his hands in front of the returned camera name (position) towards the aisle shelf. For example, the system will ask the user to move his hands in front of the top camera in the shopping carts toward the aisle shelf. The system will use skin filtering algorithms to locate the user hands coordinates. The system will guide the user based on the retrieved product location using the four directions (up, down, right, left) until the user's hands reach the product location.

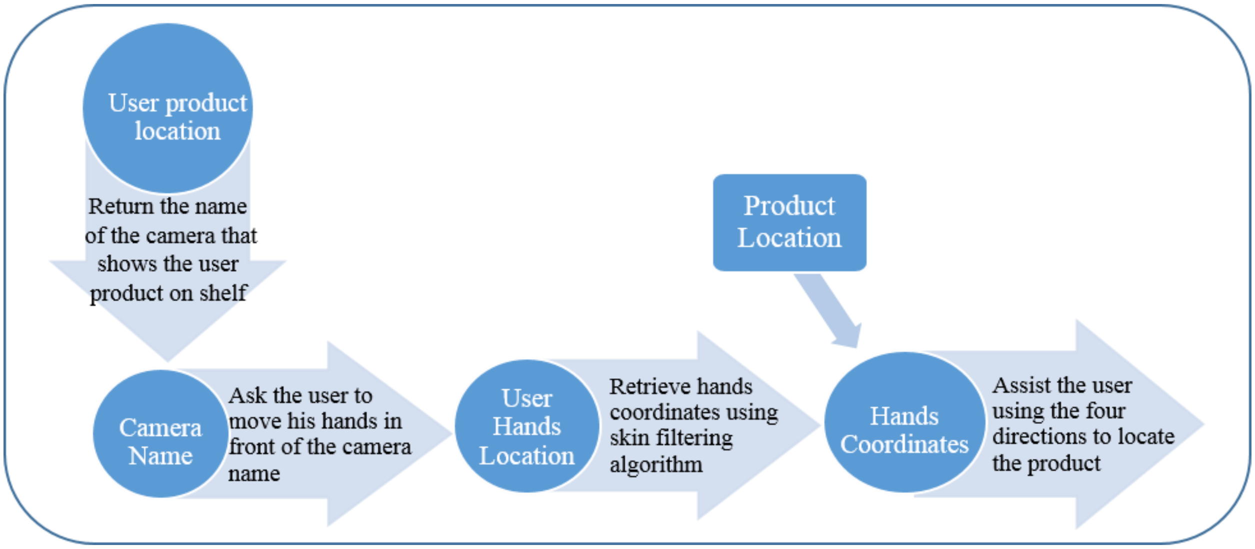

User product location
Return the name of the camera that shows the user product on shelf
Camera Name
Ask the user to move his hands in front of the camera name
User Hands Location
Retrieve hands coordinates using skin filtering algorithm
Product Location
Hands Coordinates
Assist the user using the four directions to locate the product

Figure 3.11: Guide the user to the location of the product

3.4 DataSet

We used item master dataset [41] instead of capturing product images from a real grocery store. The dataset contains more than 20,001 product images. We used 1550 images from the dataset.

The dataset contains different product images from 688 categories. Examples of category names are Milk, Cheese, Syrups, and Herbs. The dataset classifies the products based on the identification number (ID), and Universal Product Code (UPC). Each product record contains the product name, brand, manufacturer, and so on. Each product has a number of images from different views: planogram front (PF) as shown in Figure (3.12), back, top, bottom, left, and right. We used 775 PF images for IPM evaluation. The IPM method used a database contained PF view images. The PF images match in photography conditions the *in situ* images mentioned in (2.3.1.4). The *in situ* images (Figure (2.13)) shows the products photographed from the front face as the dataset PF images (Figure (2.12)).

Also, it includes product images for marketing and commercial use that are photographed from ideal views as shown in Figure (3.13). The marketing images (MK) match in photography conditions the *in vitro* images mentioned in (2.3.1.4). The *in vitro* images (Figure (2.12)) shows the products photographed isolated with transparent background and fall under ideal conditions like the MK (Figure (3.13)). The MK images are also photographed from different views: front, left, and right. We used the MK images to resemble the shelf view image instead of photographing images from real grocery store shelves. The MK images used to create montage images as shown in Figure (3.14). We used a total of 775 MK images to create 100 montage images from different 100 categories. Each montage image contained eight products MK images from the same category. We used the product name in the dataset as the user desired product name to be matched with the OCR result. The product name in the dataset contains two parts: Brand Name (BN) and Product Description (PD).For example: if the product name is Starbucks Hot Cocoa Peppermint, BN is Starbucks and PD is Hot Cocoa Peppermint. The dataset was collected using a MATLAB code to store the images from URL links typed in the dataset excel sheet along with the required informations. Furthermore, a MATLAB code was developed to create a 100 montage images from the stored product images using the montage function.

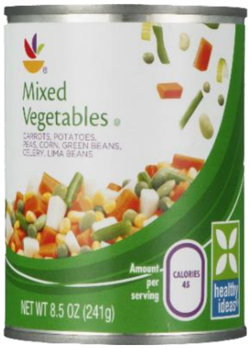


Figure 3.12: PF View product Image [41]

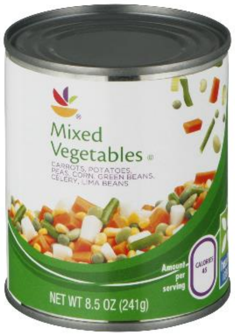


Figure 3.13: MK Product Image [41]

We grouped the dataset to four situations that the user might encounter, each containing 25 montage images:

1. If the user desired product exist in the montage image, we call the situation "*exist*".
2. If the user desired product is not exist in the montage image, we call the situation "*not exist*".

3. If a similar to the user desired product exist in the montage image, we call the situation "*similar*". The similarity may be in the BN or PD.
4. If the user desired product exist twice in the montage image, we call the situation "*twice*".

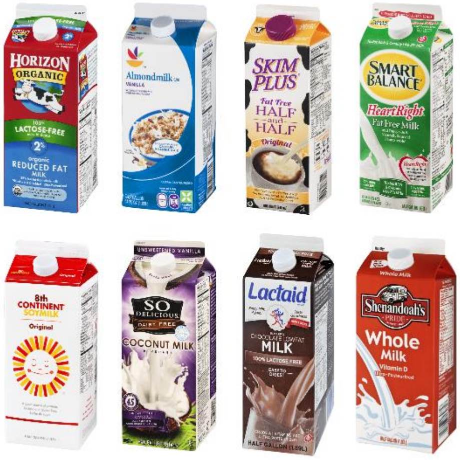


Figure 3.14: Montage Image

3.5 Proposed Analysis

In order to develop the proposed system and to apply it to the real-world, six different matching algorithms strategies were used. These strategies are:

i) IPM NN

The IPM SURF algorithm (2.3.1.2) will be used to match between the montage images and the user desired product PF images. The NN strategy mentioned in (2.3.1.3) will be used to match between the two images.

ii) IPM NNDR

The IPM SURF algorithm (2.3.1.2) will be used to match between the montage images and the user desired product PF images. The NNDR strategy mentioned in (2.3.1.3) will be used to match between the two images.

iii) OCR MK BN+PD

The ABBYY OCR engine (2.3.2.1) will be used to recognize text in the eight MK product images of one montage image. Then, match between the retrieved text of the eight images and user desired product BN and PD text.

iv) OCR PF BN+PD

The ABBYY OCR engine (2.3.2.1) will be used to recognize text in the eight PF product images of one montage image. Then, match between the retrieved text of the eight images and user desired product BN and PD text.

v) OCR MK PD

The ABBYY OCR engine (2.3.2.1) will be used to recognize text in the eight MK product images of one montage image. Then, match between the retrieved text of the eight images and user desired PD text.

vi) OCR PF PD

The ABBYY OCR engine (2.3.2.1) will be used to recognize text in the eight PF product images of one montage image. Then, match between the retrieved text of the eight images and user desired PD text.

The performance of the experiment results were measured as mentioned in Section (2.4) by calculating: Precision, Recall, Fall out, Negative Predicted Values (NPV) and Accuracy. In order to determine the latter measures, we need to calculate the following four values from the confusion matrix as was shown in Table (2.3) in the previous chapter: TP, FP, TN and FN. In this context: i) TP means that the product is available, and the system found it; ii) FP means that the product is not available, but the system found it; iii) TN means that the product is not available, and the system did not found it; iv) FN means that the product is available, but the system did not detect it.

We calculated the performance rates using the equations (2.2), (2.3), (2.4), (2.5), and (2.6). The performance rates were obtained for each dataset situation according to each one from the six algorithm strategies.

3.6 Fusion Technique

In order to construct a multimodal system of multiple algorithms used in our system, the Decision Level Fusion in parallel mode was applied using the Majority Voting method (2.5.3.2). The multimodal system starts by extracting the features either by IPM or OCR from the shelf view images. Then, the extracted features are matched with the user desired product image. Finally, the system generates a decision regarding the match. According to the matching results, the (OCR MK BN+PD) and (OCR PF BN+PD) algorithms were excluded from the fusion due to the large number of the false matching results.

The decision fusion was implemented using the majority voting equation (2.7). The number of the algorithms used for the equation is R=4. So, the equation will be implemented as follows:

$$k = \left\{ \frac{4}{2} + 1 = 3 \right. \tag{3.1}$$

Where K represents the minimum number of the algorithms that must agree on the similarity decision. Therefore, three algorithms of the four main algorithms must return a true matching result.

The decision fusion procedure used to decide the final decision with the following workflow is shown in figure (3.15):

1. Select the algorithm that collected the maximum true match.
2. Select two algorithms that have the same true match decisions such that these decisions are false decisions in the selected algorithm in Step one.
3. The algorithm in Step one is fused with the two algorithms in Step two.

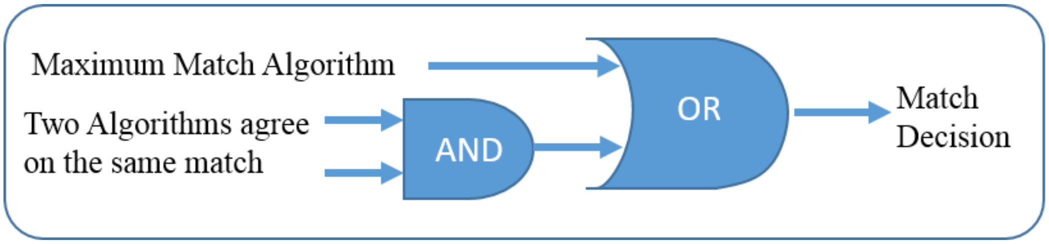


Figure 3.15: Decision Fusion Procedure

The average of the number of the match in step one was compared with the average of number of matches after fusion to compare the level of fusion improvement.

3.7 Conclusions

We want to build a device to assist the visually impaired in grocery shopping. The device consists of a shopping cart with three cameras installed on the cart. The system aims to inform the user with the name of the category he or she is located in and to advise him to locate his or her desired product on the shelf. The videos captured by the cameras on the cart will be the input data to the system. The system converts the video to a number of non-redundant images. The user will announce his or her desire product name. Then, a speech to text tool is used to convert the speech to text. The system will compare two image recognition techniques: OCR and IPM. The OCR technique recognizes the text written on the product labels to identify the user desired product. While, IPM compares the shelf view image with the required product images. The method requires building a database of product images to be compared with. The system will search the database for the product image based on the name of the user desired product. The system will be implemented on the item master dataset. The dataset contains a number of product images captured from different views. The marketing view was used to create an image resembles the grocery store shelf view image. While, the planogram front product images was used as the database required for the matching process in IPM.

Chapter 4

Results

The IPM matched the interest point features of the product images in the dataset with the images retrieved from the camera. On the other hand, OCR matched the text retrieved from the OCR engine with the name of the desired product. Each algorithm was tested under different conditions in order to ensure its correct functionality. We tested the IPM with two different matching strategies. The OCR was tested for two different image conditions and two different matching criteria. The analysis used 1550 images from the dataset belonging to 775 products. Four situations that may occur to the user in shopping were tested.

This chapter is organized as follows: Section (4.1), Introduction, presents the main aspects of the testing system; Section (4.2) Confusion Matrix, presents the confusion matrix results for testing the algorithms on the dataset and an example shows how we calculate the results; Sections (4.3) Performance Rates, presents the performance rate results of testing the algorithms on the dataset and an example shows how we calculate the results; Section (4.4) Percentage of Accuracy, presents the accuracy results in percentage especially for the *not exist* and *similar* situations followed by an example shows the calculation procedure; Section (4.5) Decision Fusion, presents the decision fusion results with a comparison between the results before and after the fusion ; and Section (4.6) Conclusion, concludes the chapter with the main points and parts of the results.

4.1 Introduction

The scope of the system is to match the user desired product with the montage image that contains eight product marketing images. The montage images used was a total of 100 image each containing eight products. The total of the products used in the analysis is 775. We used 775 marketing product images for building montage images and 775 planogram front images for OCR and IPM matching with a total of 1550 product images. The system divided the 100 montage image into the four situations that may occur such that each situation will contain 25 montage image with 200 products. The four situations that may the user face into (as presented in section (3.4)) are: *exist*, *not exist*, *similar*, and *twice*.

In the IPM algorithm, we matched the user desired product planogram front image with the montage image. The IPM algorithm was tested using two different matching strategies as described in section (2.4.1.1). The first matching strategy is called Nearest Neighbor Matching (NN). The system applied the NN: i) if the descriptor of the user desired product image is the nearest neighbor to the descriptor of the montage image; and ii) if the distance between the two descriptors is under the threshold.

The second matching strategy is Nearest Neighbor Distance Ratio Matching (NNDR). The procedure is similar to NN, but the threshold was applied to the distance ratio between the two nearest neighbors. For example, if the descriptor of the user desired product Dp and the descriptor of the montage image Dm, the NNDR strategy matched between the two descriptors if $\|Dp - D_{m1}\|/\|Dp - D_{m2}\| < t$ Such that D_{m1} is the first nearest neighbor to Dp and D_{m2} is the second nearest neighbor Dp. In both strategies, the matching were used on each descriptor of the user desired product and montage image [27]. We assigned the threshold to (0.6) as described in section (3.3.2.1).

The OCR matching was implemented using ABBYY OCR engine. To get the OCR results, the dataset images (product marketing and planogram front) were sent to the engine. In the matching stage, we matched the retrieved text of each product image from the engine with the name of the desired product mentioned in the dataset. The name of the product in the dataset contains two parts: Brand name (BN) and Product Description (PD) and the system applies the matching procedure first by the two parts and second, by the PD alone. We used threshold in the PD of the returned results equal to (0.6). If 60% only of the product description words matched with the OCR retrieved results, the match is considered true match. Finally, we will use six different methods of algorithms mentioned in (3.5).

4.2 Confusion Matrix

For each method applied, we computed the total of the confusion matrix rates: TP, TN, FP, and FN (3.6). TP and FN are always related when the product exists in the reality life. On the other hand, TN and FP are always related when the product does not exist in the reality life. The sum of the confusion matrix rates should equal to eight that is the total number of products in montage image. All the matrices in this chapter are presented in the following order: [TP TN FP FN].

After calculating the confusion matrix, the rates are divided by the scale of each value. The scale matrix is the maximum values that the confusion matrix rates can take. To get the average of each rate, each rate in the confusion matrix are summarized and divided by 25 that is the total number of montage images for each situation.

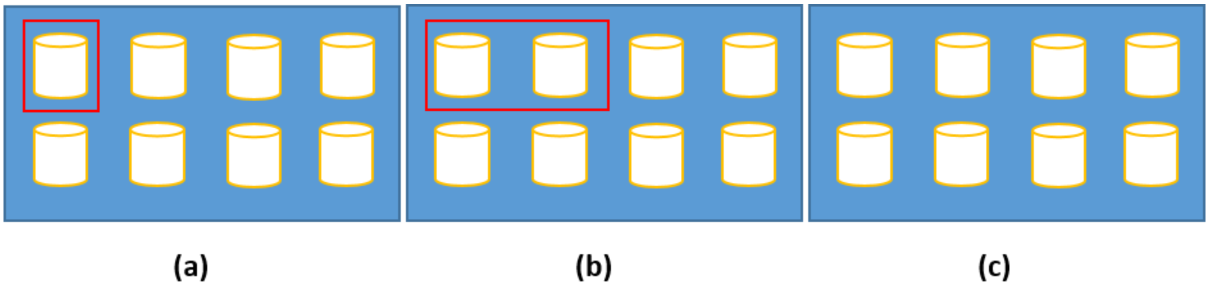


Figure 4.1: Examples of the Situations that may occur: (a) *exist* situation (b) *twice* situation (c) *not exist* and *similar* situations

In *exist* situation (Figure (4.1) (a)), the desired product matched to one product on the aisle shelve, and seven products are not matching. Therefore, the true match matrix for *exist* situation is as follows: [1 7 0 0]. In case the system results in wrong matches, TP and FN may take one of the values: {0, 1} and TN and FP may take one of the following values: {0, 1, 2, 3, 4, 5, 6, 7} with a condition that the sum of all rates should equal to eight. For example, if the system recognizes the desired product but brings one more false product as match, the confusion matrix is: [1 6 1 0]. The maximum value that TP and FN can take is one and the maximum is seven. Therefore, the scale matrix for *exist* situation is: [1 7 7 1].

In the *twice* situation (Figure (4.1) (b)), the desired product is a match to two products on the aisle shelve, and six products are not matching. Therefore, the true match matrix for the similar situation is as follows: [2 6 0 0]. In case the system results in wrong matches, TP and FN may take one of the values: {0, 1, 2} and TN and FP may take one of the following values: {0, 1, 2, 3, 4, 5, 6} with a condition that the sum of all rates should equal to eight. For example, if the system recognizes one of the desired product but does not recognize the other one, the confusion matrix will be: [1 6 0 1]. The maximum value of the TP and FN can take is two. While, the maximum value that the TN and FP can take is six. So, the scale matrix for *exist* situation is: [2 6 6 2].

In both the *not exist* and *similar* situations as (Figure (4.1) (c)), the desired product is not on the aisle shelve or a similar to the product exist on the shelves. In both situations, the desired product does not match to any product on the aisle shelve. Therefore, the values of TP and FN will not be available. The true match matrix for *not exist* and *similar* situations is as follows: [× 8 0 ×]. The (×) sign indicates that the value is not available. In case the system results in wrong matches, TN and FP may take one of the following values: {0, 1, 2, 3, 4, 5, 6, 7, 8} with a condition that the sum of all rates should equal to eight. For example, if the system recognizes one false match product as the user desired product, the confusion matrix will be [× 7 1 ×]. The maximum value that the TN and FP can take is eight. So, the scale matrix for *exist* situation is: [× 8 8 ×].

In each montage, the sum of TP and FN should equal to the number of all positives. Also, the sum of the FP and TN should equal to the number of all negatives. For example, in the *exist* situation the sum of TP and FN should equal to one, and the sum of FP and TN should equal to seven. In the *twice* situation, the sum of TP and FN should equal to two, and the sum of FP and TN should equal to six. On the other hand, in *not exist* and *similar* situations the sum of FP and TN should equal to eight.

Sometimes, the average of rates is equal to 1 or 0 and this is due to the unreliability in the OCR using brand name matching, being hard for the OCR to read the brand names of the products due to the variation in font, text color, text background and font orientation. Furthermore, the OCR may return empty results that do not contain text, and the system considers it as TN in all situations. The empty text results may cause FN in *exist* and *twice* situations only.

The average of TP, TN, FP, and FN rates were computed following the steps:

1- Calculate the value of each rate from the confusion matrix

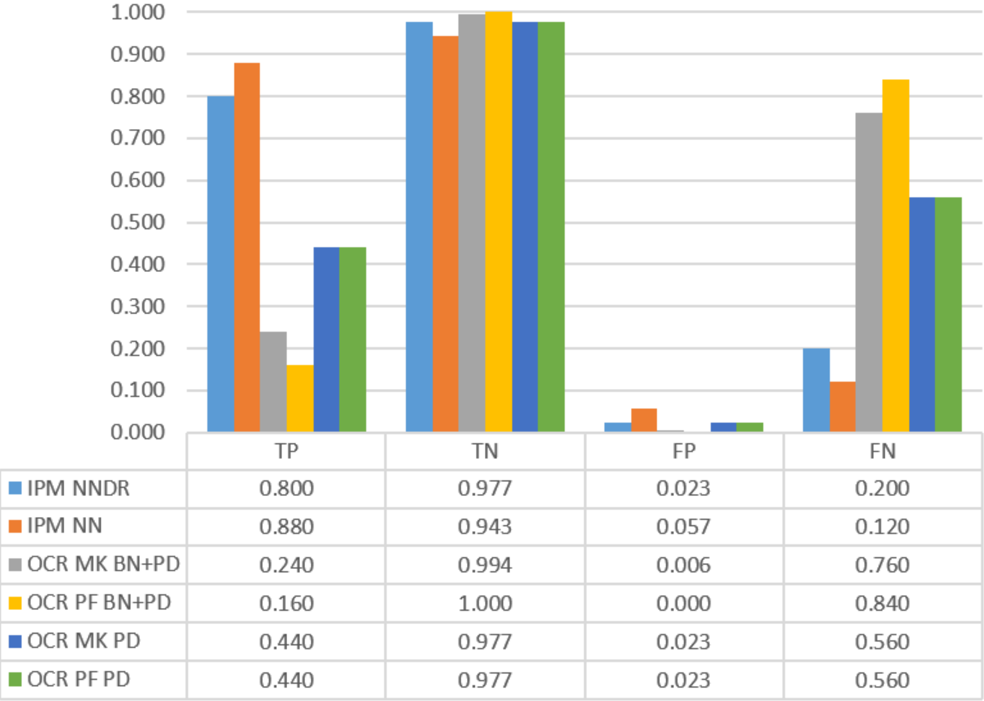
2- Divide each rate by its corresponding scale matrix.

3- Sum each rate for the 25 montage image for each situation.

4- Divide the total from Step 2 by the total montage image number that is 25.

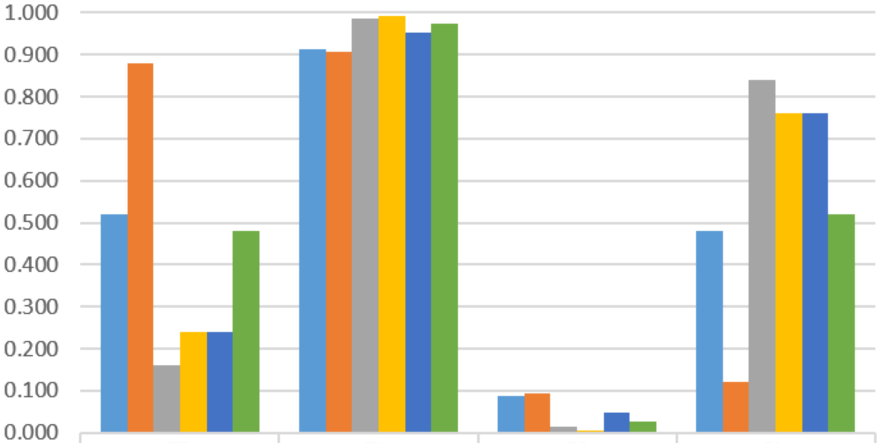
According to each algorithm, we measured the four confusion matrix averages of each situation. In *not exist* and *similar* situations, we computed only the average of TN and FP due to the unavailability of the other rates. According to each situation, Figure (4.2), (4.3), (4.4), and (4.5) shows the confusion matrix averages for each algorithm.

Example (4.1) shows how we calculated the confusion matrix rate averages for the 25 montage images in the *exist* situation.



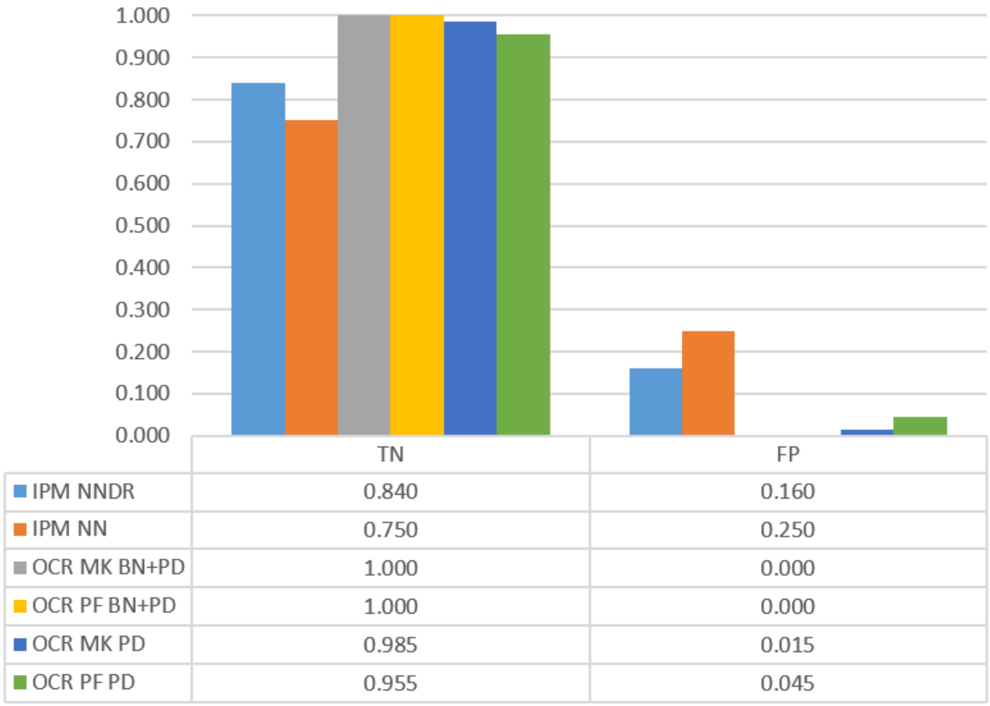
	TP	TN	FP	FN
🟦 IPM NNDR	0.800	0.977	0.023	0.200
🟧 IPM NN	0.880	0.943	0.057	0.120
⬜ OCR MK BN+PD	0.240	0.994	0.006	0.760
🟨 OCR PF BN+PD	0.160	1.000	0.000	0.840
🟦 OCR MK PD	0.440	0.977	0.023	0.560
🟩 OCR PF PD	0.440	0.977	0.023	0.560

Figure 4.2: Confusion matrix rates for *exist* situation



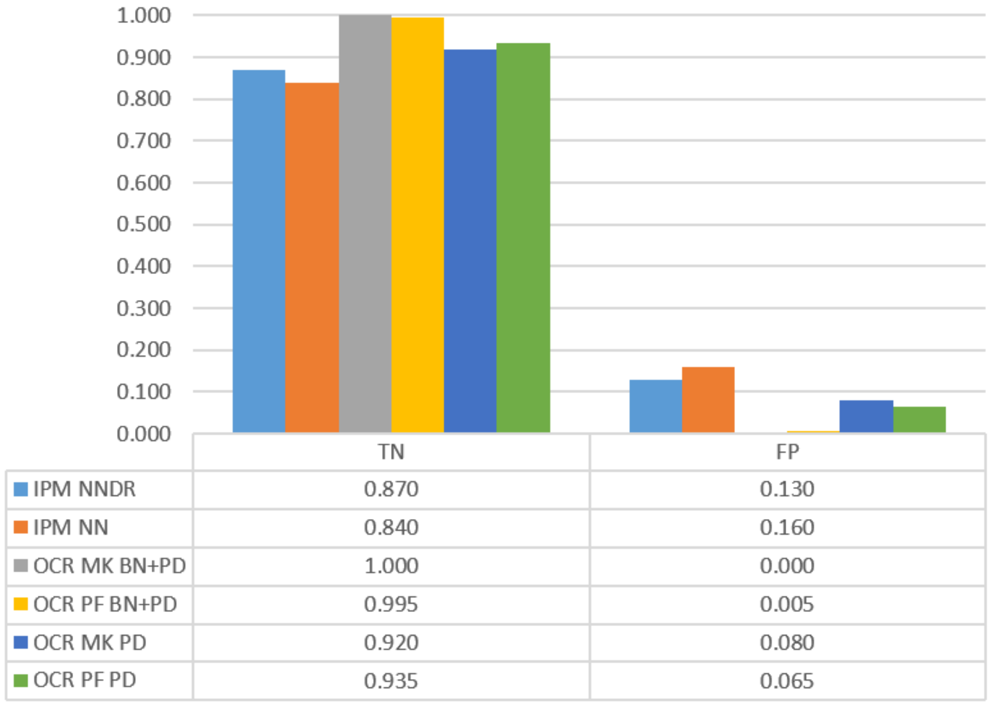
	TP	TN	FP	FN
■ IPM NNDR	0.520	0.913	0.087	0.480
■ IPM NN	0.880	0.907	0.093	0.120
■ OCR MK BN+PD	0.160	0.987	0.013	0.840
■ OCR PF BN+PD	0.240	0.993	0.007	0.760
■ OCR MK PD	0.240	0.953	0.047	0.760
■ OCR PF PD	0.480	0.973	0.027	0.520

Figure 4.3: Confusion matrix rates for *twice* situation



	TN	FP
🟦 IPM NNDR	0.840	0.160
🟧 IPM NN	0.750	0.250
◼ OCR MK BN+PD	1.000	0.000
🟨 OCR PF BN+PD	1.000	0.000
🟦 OCR MK PD	0.985	0.015
🟩 OCR PF PD	0.955	0.045

Figure 4.4: Confusion matrix rates for *not exist* situation



	TN	FP
🟦 IPM NNDR	0.870	0.130
🟧 IPM NN	0.840	0.160
◼ OCR MK BN+PD	1.000	0.000
🟨 OCR PF BN+PD	0.995	0.005
🟦 OCR MK PD	0.920	0.080
🟩 OCR PF PD	0.935	0.065

Figure 4.5: Confusion matrix rates for *similar* situation

Example (4.1):

The example was implemented on the *exist* situation for all the six algorithm. We showed how we computed the confusion matrix rates for one montage image contained different eight products from the Cakes category. The following use case is used: the user is searching for a product called Weight Watchers Chocolate Cream Cake which is located in the montage image at the bottom shelve second product from the left.

I. IPM Algorithm Strategies

Figure (4.6) shows the results of the IPM NN algorithm. In order to calculate the average of the confusion matrix rates, we started by applying the steps mentioned earlier. We applied step one and two for one montage sample (Cake category). Step three and four was applied to all the 25 montage images for *exist* situation. The described procedure was used for the whole example.

1- TP = 1, TN = 5, FP = 2, FN = 0.

2- The scale matrix for *exist* situation is: [1 7 7 1]

TP = 1/1 = 1, TN = 5/7 = 0.714, FP = 2/7 = 0.286, FN = 0/1= 0

3- TP = 22, TN = 23.571 , FP = 1.429, FN = 3

4- TP = 22/25= 0.880, TN = 23.571/25= 0.9943, FP = 1.429/25= 0.057, FN = 3/25= 0.120

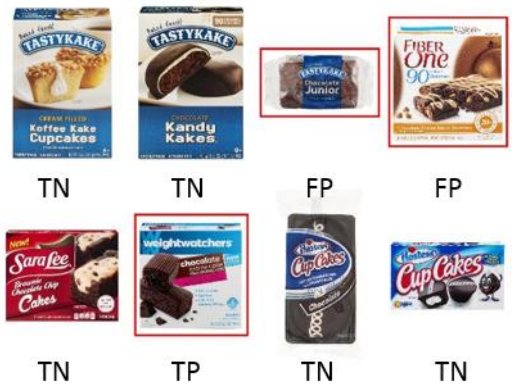


Figure 4.6: IPM NN results of cake category image

Figure (4.7) shows the results of the IPM NNDR. In order to calculate the average of the confusion matrix rates, we started by applying the steps mentioned earlier.

1- TP = 0, TN = 6, FP = 1, FN = 1.

2- The scale matrix for *exist* situation is: [1 7 7 1]

   TP = 0/1 = 0, TN = 6/7 = 0.857, FP = 1/7 = 0.143, FN =1/1 = 1.

3- TP = 20, TN = 24.429, FP = 0.571, FN = 5

4- TP = 20/25 = 0.800, TN = 24.429/25 = 0.977, FP = 0.571/25 = 0.023, FN = 5/25 = 0.200

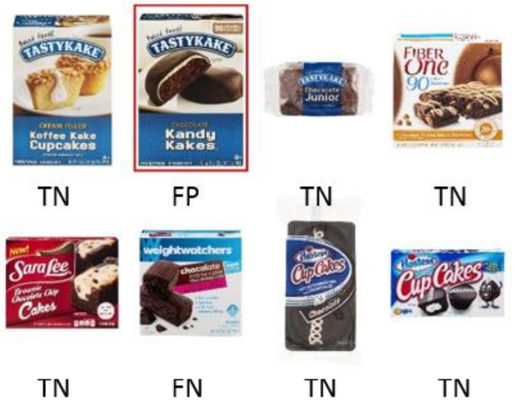


Figure 4.7: IPM NNDR results of cake category image

Table (4.4) shows the confusion matrix rates of all the 25 montage images for *exist* situation using OCR algorithms.

Table 4.1: Confusion Matrix Rates for *exist* situation using IPM algorithms

Image	IPM NNDR				IPM NN			
	TP	**TN**	**FP**	**FN**	**TP**	**TN**	**FP**	**FN**
1	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
2	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
3	0.000	0.857	0.143	1.000	0.000	0.857	0.143	1.000
4	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
5	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
6	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
7	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
8	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
9	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
10	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
11	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
12	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
13	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
14	0.000	0.857	0.143	1.000	1.000	0.714	0.286	0.000
15	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
16	0.000	1.000	0.000	1.000	1.000	0.714	0.286	0.000
17	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
18	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
19	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
20	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
21	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
22	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
23	0.000	0.857	0.143	1.000	0.000	0.571	0.429	1.000
24	0.000	0.857	0.143	1.000	0.000	0.714	0.286	1.000
25	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
Sum	20.000	24.429	0.571	5.000	22.000	23.571	1.429	3.000
Average	**0.800**	**0.977**	**0.023**	**0.200**	**0.880**	**0.943**	**0.057**	**0.120**

II. OCR Algorithm Methods

The name of the product that the user is searching for in the cake category image is Weight Watchers Chocolate Cream Cake. The desired product is the 6th product in the montage image. The BN is Weight Watchers, and the PD is Chocolate Cream Cake. When comparing with the BN and PD, the BN words (Ex. Weight Watchers) must result from OCR and PD words (Ex. Chocolate Cream Cake) must exist using threshold in the returned result from OCR. While, in comparing with the PD, only the PD words must exist in the OCR result using the threshold. We used threshold (0.6), and the PD is three words. Therefore, three words × 0.6 threshold =1.8 ≈ two words. As a result, the minimum number of words that must exist in each OCR result of a product description is two words. For example, if Chocolate Cream exist in the returned result from OCR, it considered a match.

Table (4.2) shows the results of the OCR MK algorithm matching with the BN and PD. In order to calculate the average of the confusion matrix rates, we started by applying the steps mentioned earlier.

1- TP = 1, TN = 7, FP = 0, FN = 0.

2- The scale matrix for exist situation is: [1 7 7 1]

TP= 1/1 = 1, TN = 7/7 = 1, FP = 0/7=0, FN = 0/1=0.

3- TP = 6, TN = 24.857, FP = 0.143, FN = 19

4- TP = 6/25 = 0.240, TN = 24.857/25 = 0.994, FP = 0.143/25 = 0.006, FN = 19/25 = 0.760.

The results of the OCR MK algorithm matching with the PD alone shows the same result when matching with the BN and PD.

Table 4.2: OCR MK algorithm results of Cake category image

Shelve order	**Products on First Shelf (from left)**				**Products on Second Shelf (from left)**			
Product order	**1st**	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**
Match BN+PD	TN	TN	TN	TN	TN	TP	TN	TN
Match PD	TN	TN	TN	TN	TN	TP	TN	TN
	Product enlarged to show texture CREAM FILLED Koffee Kake Cupcakes 12 CRUMB TOPPED CUPCAKES _ ©D family pack 16 packs of 2 NET WT1275 OZ (361g) 6-2V8 OZ PKGS		'NlRSi OilUH uju	6 ■ 0.89 OZ (25g) BROW NIES NET WT 5,34 OZ (151 g) ©d PER BROW NIE	garajez Chip Cho^ri . Cjok** 7 INDIVIDUALL Y WRAPPED CAKES * ^Baked^ PRODUCT ENLARGED TO SHOW TEXTURE PERI CAKE SERVING 12.25 OZ (347g)	jOUSCOVfi' p Enlarged to Show Texture Serving Suggestion weight watchers + 90 calories + 3g fiber * rich and cream filling per cake 6 cakes individually wrapped NET WT 5.7 0Z(162g)		

Table (4.3) shows the results of the OCR PF algorithm matching with the BN and PD.

1- TP = 0, TN = 7, FP = 0, FN = 1.

2- The scale matrix for exist situation is: [1 7 7 1]

TP = 0/1 = 0, TN = 7/7 = 1, FP = 0/7 = 0, FN= 1/1 = 1.

3- TP = 4, TN = 25, FP =0, FN = 21

4- TP = 4/25 = 0.160, TN = 25/25 = 1, FP =0/25 = 0, FN = 21/25 = 0.840

Table 4.3: OCR PF Results of Cake category image

Shelve order	Products on First Shelf (from left)				Products on Second Shelf (from left)			
Product order	1st	2nd	3rd	4th	5th	6th	7th	8th
Match BN+PD	TN	TN	TN	TN	TN	FN	TN	TN
Match PD	TN	FP	TN	TN	TN	TP	TN	TN
	CREAM FILLED Koffee Kake Cupcakes 12 CRUMB TOPPED CUPCAKES ^ ©D family pack 16 packs of 2 NETWT 12.75 OZ (361g) 6-21/8 OZ PKGS	QA CALORIES 7U PER CAKE! Product enlarged to show texture CHOCOLATE Kandy Kakes (B)d family pack 16 packs of 2 NET.WT8 OZ (227g) 6-1]h OZ PKGS			0 Saram ■* ?/cW2t. Chip Baked •fresh • PRODUCT ENLARGED to go! TO SHOW TEXTURE PERI CAKE SERVING 7 INDIVIDUALLY wrapped cakes 12.25 0Z(347g)	+ 90 calories + 3g fiber + rich and cream filling per cake 6 cakes individually wrapped ©D NET WT 5.7 0Z|162g) Enlarged to Show Texture Serving Suggestion		

Furthermore, Table (4.3) shows the results of the OCR PF algorithm matching with the PD alone. In order to calculate the average of the confusion matrix rates, we started by applying the steps mentioned earlier.

1- TP =1, TN = 6, FP = 1, FN = 0

2- The scale matrix for exist situation is: [1 7 7 1]

TP = 1/1 = 1, TN = 6/7 = 0.857, FP = 1/7 = 0.142, FN = 0/1 = 0

3- TP = 11, TN = 24.429, FP = 0.561, FN = 14

4- TP = 11/25 = 0.440, TN = 24.429/25 = 0.977, FP = 0.561/25 = 0.023, FN = 14/25 = 0.560

Table (4.4) shows the confusion matrix rates of all the 25 montage images for *exist* situation using OCR algorithms. The results of the averages mentioned in the example were represented earlier in Figure (4.2).

Table 4.4: Confusion Matrix Rates for *exist* situation using OCR algorithms

#	OCR MK BN+PD				OCR MK PD				OCR PF BN+PD				OCR PF PD			
	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN	TP	TN	FP	FN
1	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
2	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
3	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000
4	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
5	0.000	1.000	0.000	1.000	0.000	0.857	0.143	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
6	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
7	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
8	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
9	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000
10	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
11	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
12	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
13	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
14	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	1.000	0.857	0.143	0.000
15	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
16	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000	1.000	1.000	0.000	0.000
17	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000
18	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
19	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000
20	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000	0.000	1.000	0.000	1.000	1.000	1.000	0.000	0.000
21	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
22	0.000	0.857	0.143	1.000	1.000	0.857	0.143	0.000	0.000	1.000	0.000	1.000	1.000	0.857	0.143	0.000
23	0.000	1.000	0.000	1.000	0.000	0.857	0.143	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000
24	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	0.857	0.143	1.000
25	0.000	1.000	0.000	1.000	0.000	0.857	0.143	1.000	0.000	1.000	0.000	1.000	0.000	0.857	0.143	1.000
Sum	6.000	24.857	0.143	19.000	11.000	24.429	0.571	14.000	4.000	25.000	0.000	21.000	11.000	24.429	0.571	14.000
Average	0.240	0.994	0.006	0.760	0.440	0.977	0.023	0.560	0.160	1.000	0.000	0.840	0.440	0.977	0.023	0.560

4.3 Performance Rates

We measured the performance rates as described in section (3.4) using: TPR, PPV FPR, NPV, and ACC. The equations of each rate were given in section (2.4).

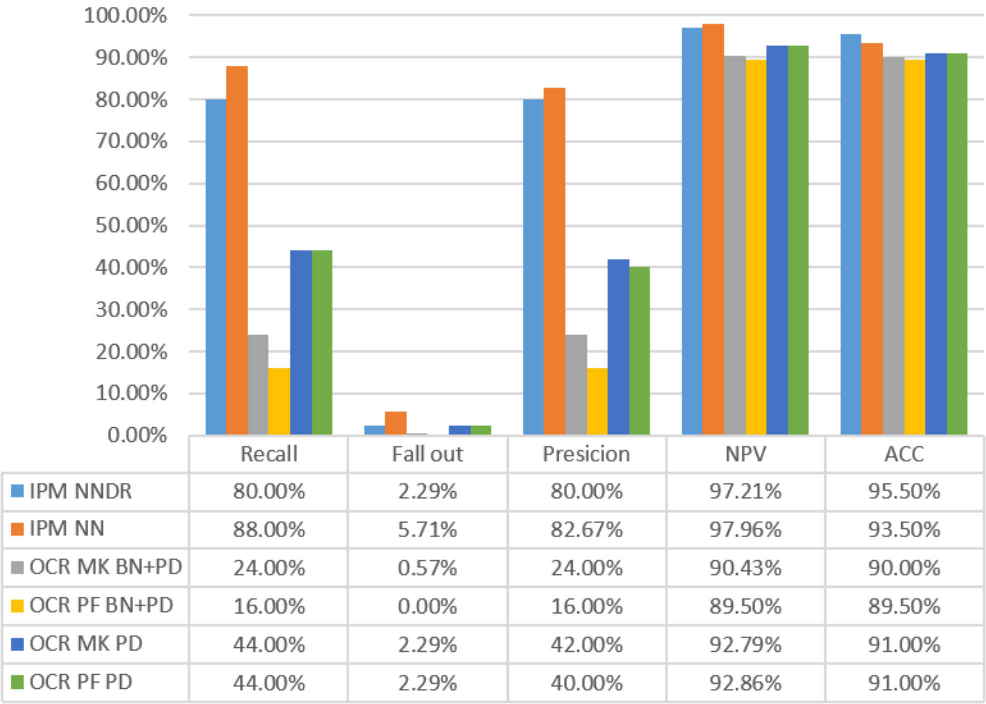
For each situation, we calculated each performance rate for the 25 montage image using the equations. Then, we summed the results and divided them by 25. In the *not exist* and *similar* situations, we did not calculate the recall and precision because TP and FN are available (as mentioned in (4.2)). The simulation calculated the accuracy in *not exist* and *similar* situations by omitting the TP and FN as follows:

$$ACC = \frac{TN}{TN+FP} \tag{4.1}$$

In *not exist* and *similar* situations, the NPV is equal to one because the value of FN is not available. TN will never be equal to zero because the system should recognize at least one of the TNs. For that, the value of the NPV will not be shown in the *not exist* and *similar* situations figures. NPV equation will be:

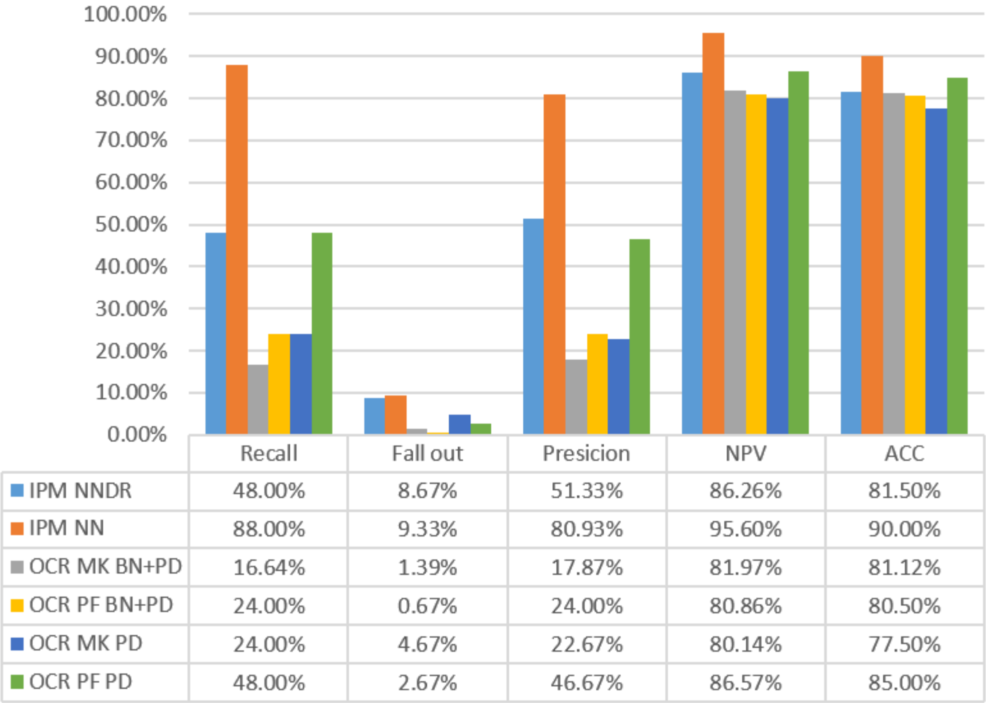
$$NPV = \frac{TN}{TN} = 1 \tag{4.2}$$

In case we face the problem of division by zero, the result is excluded from the summation of the 25 images. According to each situation, Figure (4.8), (4.9), (4.10), and (4.11) shows the performance rates for each algorithm.



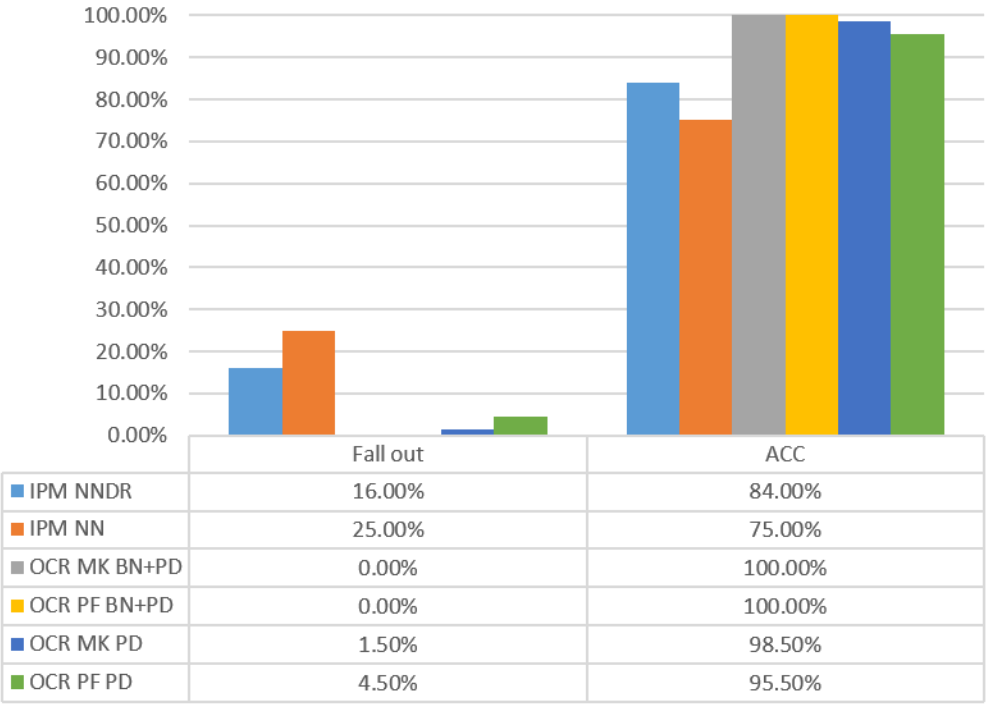
	Recall	Fall out	Presicion	NPV	ACC
🟦 IPM NNDR	80.00%	2.29%	80.00%	97.21%	95.50%
🟧 IPM NN	88.00%	5.71%	82.67%	97.96%	93.50%
◼ OCR MK BN+PD	24.00%	0.57%	24.00%	90.43%	90.00%
🟨 OCR PF BN+PD	16.00%	0.00%	16.00%	89.50%	89.50%
🟦 OCR MK PD	44.00%	2.29%	42.00%	92.79%	91.00%
🟩 OCR PF PD	44.00%	2.29%	40.00%	92.86%	91.00%

Figure 4.8: Performance rates for *exist* situation



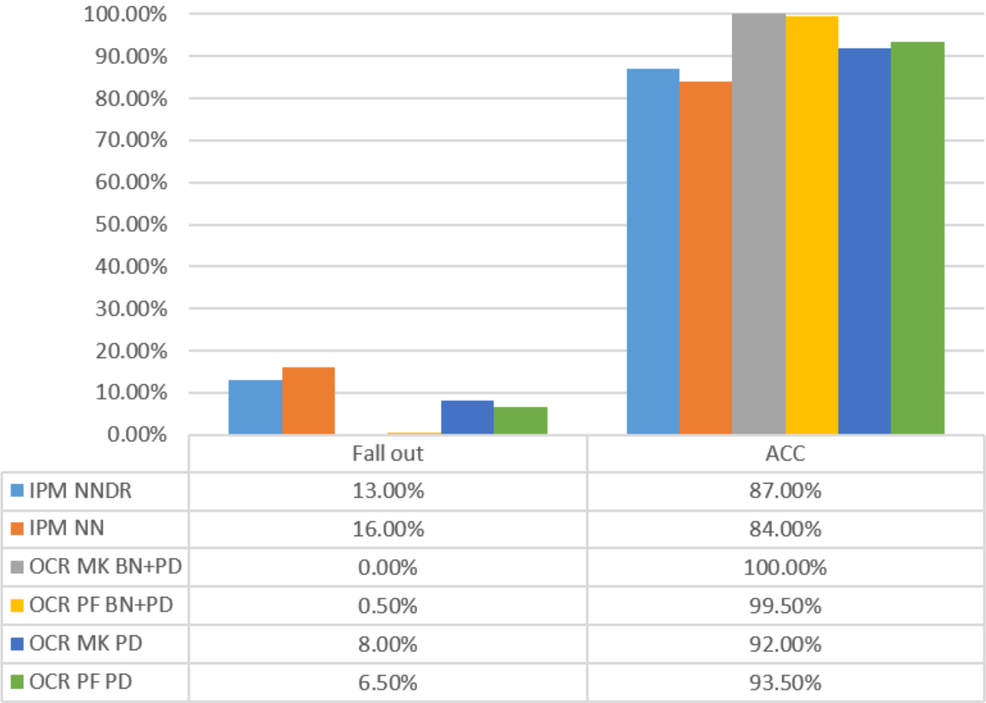
	Recall	Fall out	Presicion	NPV	ACC
🟦 IPM NNDR	48.00%	8.67%	51.33%	86.26%	81.50%
🟧 IPM NN	88.00%	9.33%	80.93%	95.60%	90.00%
◼ OCR MK BN+PD	16.64%	1.39%	17.87%	81.97%	81.12%
🟨 OCR PF BN+PD	24.00%	0.67%	24.00%	80.86%	80.50%
🟦 OCR MK PD	24.00%	4.67%	22.67%	80.14%	77.50%
🟩 OCR PF PD	48.00%	2.67%	46.67%	86.57%	85.00%

Figure 4.9: Performance rates for *twice* situation



	Fall out	ACC
■ IPM NNDR	16.00%	84.00%
■ IPM NN	25.00%	75.00%
■ OCR MK BN+PD	0.00%	100.00%
■ OCR PF BN+PD	0.00%	100.00%
■ OCR MK PD	1.50%	98.50%
■ OCR PF PD	4.50%	95.50%

Figure 4.10: Performance rates for *not exist* situation



	Fall out	ACC
■ IPM NNDR	13.00%	87.00%
■ IPM NN	16.00%	84.00%
■ OCR MK BN+PD	0.00%	100.00%
■ OCR PF BN+PD	0.50%	99.50%
■ OCR MK PD	8.00%	92.00%
■ OCR PF PD	6.50%	93.50%

Figure 4.11: Performance rates for *similar* situation

Example (4.2) will show how we calculated the performance rates for exist situation using the IPM NNDR algorithm. The results were represented in Figure (4.8).

Example (4.2):

We will calculate the average of the Recall, Precision, Fall out, NPV and ACC for exist situation using IPM NNDR algorithm. In order to calculate the average, we first calculate each one of the performance rates for each one from the 25 montage images. The inputs for the equations are the confusion matrix rates that were calculated in Example (4.1) and shown in Table (4.1). Then, we sum all the results for each performance rate and divide it by 25. The results are shown in Table (4.5).

Table 4.5: Performance Rates for *exist* situation using IPM NNDR algorithm

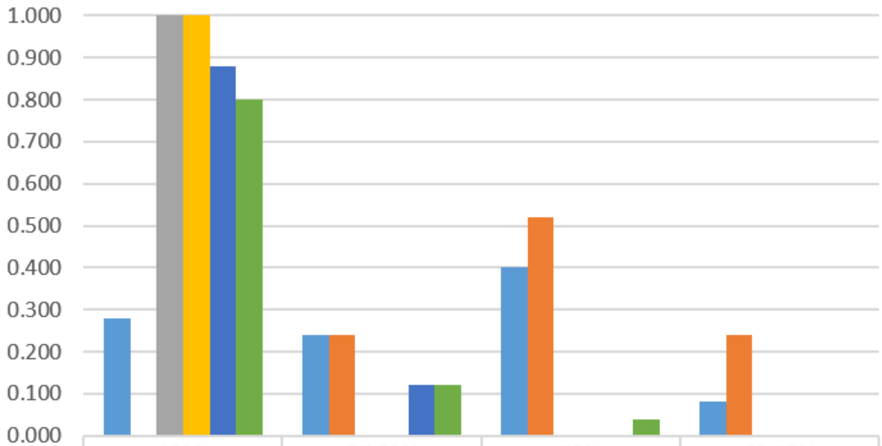
Image	Recall	Fall out	Precision	NPV	ACC
1	1.000	0.000	1.000	1.000	1.000
2	1.000	0.000	1.000	1.000	1.000
3	0.000	0.143	0.000	0.857	0.750
4	1.000	0.000	1.000	1.000	1.000
5	1.000	0.000	1.000	1.000	1.000
6	1.000	0.000	1.000	1.000	1.000
7	1.000	0.000	1.000	1.000	1.000
8	1.000	0.000	1.000	1.000	1.000
9	1.000	0.000	1.000	1.000	1.000
10	1.000	0.000	1.000	1.000	1.000
11	1.000	0.000	1.000	1.000	1.000
12	1.000	0.000	1.000	1.000	1.000
13	1.000	0.000	1.000	1.000	1.000
14	0.000	0.143	0.000	0.857	0.750
15	1.000	0.000	1.000	1.000	1.000
16	0.000	0.000	0.000	0.875	0.875
17	1.000	0.000	1.000	1.000	1.000
18	1.000	0.000	1.000	1.000	1.000
19	1.000	0.000	1.000	1.000	1.000
20	1.000	0.000	1.000	1.000	1.000
21	1.000	0.000	1.000	1.000	1.000

22	1.000	0.000	1.000	1.000	1.000
23	0.000	0.143	0.000	0.857	0.750
24	0.000	0.143	0.000	0.857	0.750
25	1.000	0.000	1.000	1.000	1.000
Sum	20.000	0.571	20.000	24.304	23.875
Average	**80.00%**	**2.29%**	**80.00%**	**97.21%**	**95.50%**

4.4 Percentage of Accuracy

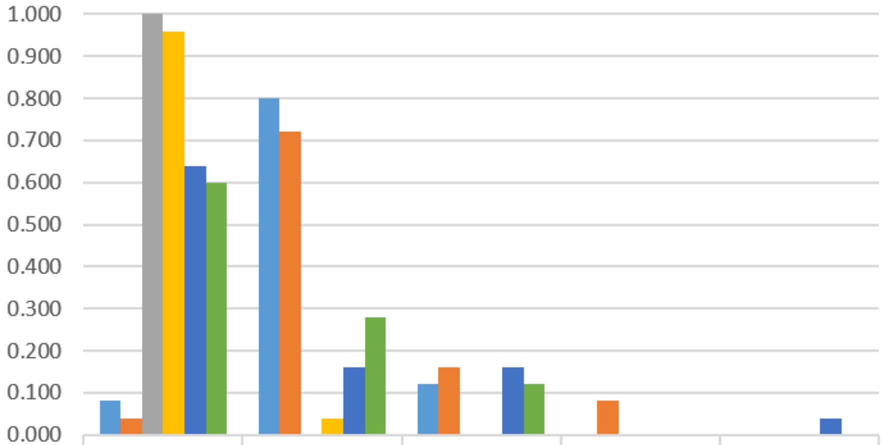
In case of *not exist* and *similar* situations, the value of precision and recall cannot be detected. Therefore, we used the percentage of accuracy. In both cases, the product does not exist in the montage image that mean the value of TN should equal to eight and FP should equal to zero. There are eight levels of accuracy. Each higher level is less than 12.5% than the lower level. If the system detected the TN as eight and FP as zero, then the system is 100% accurate. If the system detected the TN as seven and FP as one, then the system is 87.5% accurate. If the system detected the TN as six and FP as two, then the system is 75% accurate. If the system detected the TN as five and FP as three, then the system is 62.55% accurate. If the system detected the TN as four and FP as four, then the system is 50% accurate. The previous TN and FP values are the only values that the system resulted in our experiment. Then, we counted how many montage images obtained the same percentage of accuracy. Finally, divided the number of montage images by 25 to get the average.

The percentage of accuracy indicates the percentage of the number of montage images that calculated the confusion matrix rates accurately. Figure (4.12) shows the accuracy percentage of *not exist* situation according to each algorithm results. Figure (4.13) shows the accuracy percentage of the *similar* situation according to each algorithm results. Example (4.3) shows how we calculated the percentage of accuracy for a *similar* situation using OCR MK PD algorithm.



	100%	87.50%	75%	62.55%
■ IPM NNDR	0.280	0.240	0.400	0.080
■ IPM NN	0.000	0.240	0.520	0.240
■ OCR MK BN+PD	1.000	0.000	0.000	0.000
■ OCR PF BN+PD	1.000	0.000	0.000	0.000
■ OCR MK PD	0.880	0.120	0.000	0.000
■ OCR PF PD	0.800	0.120	0.040	0.000

Figure 4.12: Accuracy percentage of *not exist* situation



	100%	87.50%	75%	62.55%	50%
■ IPM NNDR	0.080	0.800	0.120	0.000	0.000
■ IPM NN	0.040	0.720	0.160	0.080	0.000
■ OCR MK BN+PD	1.000	0.000	0.000	0.000	0.000
■ OCR PF BN+PD	0.960	0.040	0.000	0.000	0.000
■ OCR MK PD	0.640	0.160	0.160	0.000	0.040
■ OCR PF PD	0.600	0.280	0.120	0.000	0.000

Figure 4.13: Accuracy percentage of *similar* situation

Example (4.3):

We will explain how we calculated the percentage of accuracy for a similar situation using OCR MK PD algorithm. In *not exist* situation, the product is not exist in the montage image. Thus, the system is considered 100% accurate if the TN equals to eight that mean no image from the eight images in the montage is true positive, and the FP is zero. The other levels of accuracy are decreased by 12.5 % than the higher levels. We obtained the following procedure to count the percentage of accuracy. The results of the example are shown in Figure (4.13).

1- Count how many montage images resulted from the 25 montage images with a TN=8 and FP =0. The number of the counted images are 100% accurate montage images.

2- Count how many montage images resulted from the 25 montage images with a TN=7 and FP =1. The number of the counted images are the 87.5% accurate montage images.

3- Count how many montage images resulted from the 25 montage images with a TN=6 and FP =2. The number of the counted images are the 75% accurate montage images.

4- Count how many montage images resulted from the 25 montage images with a TN=5 and FP =3. The number of the counted images are the 62.55% accurate montage images.

5- Count how many montage images resulted from the 25 montage images with a TN=4 and FP =4. The number of the counted images are the 50% accurate montage images.

6- To obtain the average, divide each result from step one, two, three, four and five by 25 that is the total number of the montage images used for the experiment.

7- Show each result after division with its corresponding percent of accuracy as shown in Table (4.6).

Table 4.6: Percentage of accuracy for *similar* situations using OCR MK PD

Percent of Accuracy	TN	FP	Number of Montage Images	Average
100%	8	0	16	**0.64**
87.5%	7	1	4	**0.16**
75%	6	2	4	**0.16**
62.55%	5	3	0	**0**
50%	4	4	1	**0.04**

4.5 Decision Fusion

As discussed in (3.6), we used the Decision Fusion technique to create a multimodal system of the six algorithms used in the current thesis. We made the fusion for the four situations each with a 25 montage image. The OCR matching using the BN and PD in both the MK and PF images in *exist* and *twice* situations was excluded due to the large number of false matching results such that the OCR cannot recognize the products BN. The algorithms options to fuse in *exist*, and *twice* situations are IPM NN, IPM NNDR, OCR MK PD, and OCR PF PD.

The *not exist* and *similar* situations, cannot fused due to large number of the false match resulted from comparing with IPM. The reason behind the large number of the false match in the IPM algorithm is that the percentage of accuracy is too low. Furthermore, due to the large number of empty texts recognized as TNs resulted from comparing with OCR matching using BN and PD. The algorithm used for decision fusion for each situation is explained as below:

1- Defining which montage image from the 25 montage images obtained a true match. The system considers a match as a true match if the values of the confusion matrix rates equal to the true match matrix-explained in section (4.2). Else, the match is considered the false match.

2- Count the total number of the true match for each algorithm and divide it by 25 to get the average.

3- Select the algorithm that have the higher matching averages. If the there are two algorithms obtained the same number of the higher true match, one of them selected.

4- Search for two algorithms that have similar true matching results such that these matching results are false matches for the algorithm is Step 3.

5- The fusion is implemented between the algorithm in Step 3 and the two algorithms in Step 4.

Tables (4.7) and (4.8) show the fusion procedure for *exist* and *similar* situations. Figures (4.11) and (4.12) show the fusion results for *exist* and *similar* situations.

Table 4.7: Fusion procedure for *exist* situation

Exist Situation	**Step 3**		**Step 4**		**Step 5**
Montage	**IPM NNDR**	**IPM NN**	**OCR MK PD**	**OCR PF PD**	**Fusion Result**
1	TRUE	TRUE	TRUE	TRUE	TRUE
2	TRUE	TRUE	FALSE	FALSE	TRUE
3	FALSE	FALSE	TRUE	TRUE	**TRUE**
4	TRUE	TRUE	FALSE	FALSE	TRUE
5	TRUE	TRUE	FALSE	FALSE	TRUE
6	TRUE	TRUE	FALSE	FALSE	TRUE
7	TRUE	TRUE	TRUE	FALSE	TRUE
8	TRUE	TRUE	FALSE	FALSE	TRUE
9	TRUE	TRUE	TRUE	TRUE	TRUE
10	TRUE	TRUE	TRUE	FALSE	TRUE
11	TRUE	TRUE	FALSE	TRUE	TRUE
12	TRUE	TRUE	FALSE	TRUE	TRUE
13	TRUE	TRUE	FALSE	FALSE	TRUE
14	FALSE	FALSE	TRUE	FALSE	FALSE
15	TRUE	TRUE	FALSE	FALSE	TRUE
16	FALSE	FALSE	TRUE	TRUE	**TRUE**
17	TRUE	TRUE	FALSE	TRUE	TRUE
18	TRUE	TRUE	TRUE	FALSE	TRUE

19	TRUE	TRUE	TRUE	TRUE	TRUE
20	TRUE	TRUE	TRUE	TRUE	TRUE
21	TRUE	TRUE	FALSE	FALSE	TRUE
22	TRUE	TRUE	FALSE	FALSE	TRUE
23	FALSE	FALSE	FALSE	FALSE	FALSE
24	FALSE	FALSE	FALSE	FALSE	FALSE
25	TRUE	TRUE	FALSE	FALSE	TRUE
Total True Match	20	20	10	9	22
Average	**80.00%**	80.00%	40.00%	36.00%	**88.00%**

Table 4.8: Fusion Procedure for *Twice* Situation

Exist Situation	**Step 3**		**Step 4**		**Step 5**
Montage	**IPM NNDR**	**IPM NN**	**OCR MK PD**	**OCR PF PD**	**Fusion Result**
1	TRUE	TRUE	FALSE	FALSE	TRUE
2	TRUE	TRUE	FALSE	TRUE	TRUE
3	FALSE	TRUE	TRUE	TRUE	TRUE
4	FALSE	TRUE	FALSE	FALSE	TRUE
5	FALSE	FALSE	FALSE	FALSE	FALSE
6	FALSE	TRUE	FALSE	FALSE	TRUE
7	FALSE	FALSE	FALSE	TRUE	FALSE
8	FALSE	FALSE	FALSE	TRUE	FALSE
9	FALSE	FALSE	FALSE	FALSE	FALSE
10	FALSE	TRUE	FALSE	FALSE	TRUE
11	TRUE	TRUE	FALSE	FALSE	TRUE
12	FALSE	FALSE	FALSE	FALSE	FALSE
13	FALSE	FALSE	FALSE	FALSE	FALSE
14	TRUE	TRUE	FALSE	FALSE	TRUE
15	FALSE	TRUE	TRUE	TRUE	TRUE
16	TRUE	TRUE	FALSE	FALSE	TRUE
17	FALSE	FALSE	FALSE	TRUE	FALSE
18	FALSE	TRUE	TRUE	TRUE	TRUE
19	TRUE	TRUE	FALSE	TRUE	TRUE
20	TRUE	TRUE	FALSE	FALSE	TRUE
21	FALSE	TRUE	FALSE	FALSE	TRUE
22	FALSE	FALSE	FALSE	TRUE	FALSE
23	FALSE	FALSE	TRUE	TRUE	**TRUE**
24	FALSE	TRUE	TRUE	TRUE	TRUE

25	TRUE	TRUE	FALSE	FALSE	TRUE
Total True Match	8	16	5	11	17
Average	32.00%	**64.00%**	20.00%	44.00%	**68.00%**

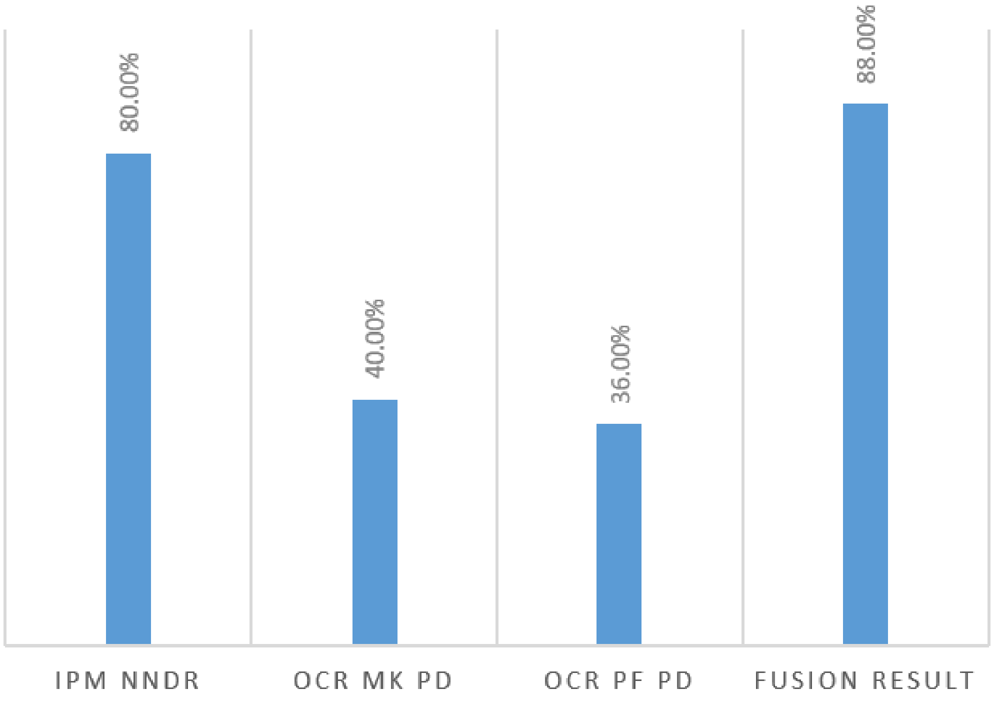


Figure 4.14: Fusion results for *exist* situation

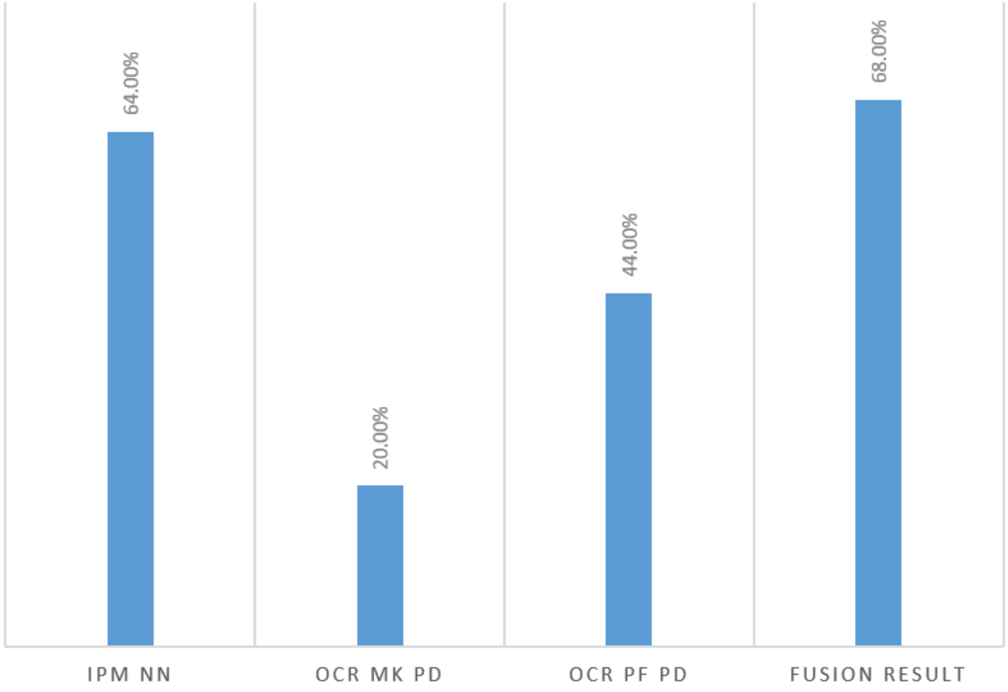


Figure 4.15: Fusion results for *twice* situation

Example (4.4) clarifies the fusion procedure used for exist situation.

Example (4.4):

In *exist* situation, the desired product exists in the montage image. Thus, the scale of the matched sample is TP=1, TN=7, FP=0, and FN=0. This example shows the results obtained with following algorithms IPM NNDR, IPM NN, OCR MK PD, and OCR PF PD by using the fusion procedure steps explained in section (4.5):

1- We showed Step one only for IPM NNDR. The true match scale for *exist* situation is [1 7 0 0]. Thus, we search for the true match matrix in the 25 montage image confusion matrix results shown in Table (4.9).

2- We counted the number of the true matches and then divide it by 25.The total true match for IPM NNDR is 20 and divided by 25 is equal 0.8. We used the average for the true match in percentage to compare it with the result of fusion.

3- The algorithms that obtained maximum true values as shown in Table (4.7) are: IPM NNDR and IPM NN. So, we select randomly one of them that is IPM NNDR.

4- The two algorithms that have similar true matching results such that these matching results are false matches for algorithm IPM NNDR is OCR MK PD and OCR PF PD as shown in Table (4.7)

5- The fusion was made between IPM NNDR, OCR MK PD, and OCR PF PD.

Table 4.9: Confusion Matrix Rates comparison with the True Match Matrix for *exist* situation using IPM NNDR

Images	**TP**	**TN**	**FP**	**FP**	**True Match Matrix [1 7 0 0]**
1	1	7	0	0	**True**
2	1	7	0	0	**True**
3	0	6	1	1	**False**
4	1	7	0	0	**True**
5	1	7	0	0	**True**
6	1	7	0	0	**True**
7	1	7	0	0	**True**
8	1	7	0	0	**True**
9	1	7	0	0	**True**
10	1	7	0	0	**True**
11	1	7	0	0	**True**
12	1	7	0	0	**True**
13	1	7	0	0	**True**
14	0	6	1	1	**False**
15	1	7	0	0	**True**
16	0	7	0	1	**False**
17	1	7	0	0	**True**
18	1	7	0	0	**True**
19	1	7	0	0	**True**
20	1	7	0	0	**True**
21	1	7	0	0	**True**
22	1	7	0	0	**True**
23	0	6	1	1	**False**
24	0	6	1	1	**False**
25	1	7	0	0	**True**

4.6 Conclusion

We tested the system using the IPM algorithm under two strategies: NN and NNDR. Furthermore, we experimented the system using OCR algorithm in two dataset images criteria: MK and PF. The OCR uses two matching techniques: matching with the BN and PD, and matching with PD alone. The dataset used had 1550 images including the marketing and the planogram front images. The montage images was a total of 100 images. The 100 montage images were divided into four situations that may occur in the system. These are *exist*, *not exist*, *similar*, and *twice*. We measured the confusion matrix rates, the performance measures, and the percentage of accuracy. These measures were calculate according to each algorithm in each situation. The confusion matrix rates measured the number of the true match products, true nonmatch products, false match products, and false nonmatch products. The performance rates measured the Precision, Recall, Fall out, NPV, and Accuracy. The percentage of accuracy help us indicates the accuracy in case of *not exist* and *similar* situations. Chart diagrams used in the chapter to emphasize the results of the rates measures in the chapter. In order to enhance the performance, we used the Decision Fusion techniques to create a multimodal system constructing from algorithms used in the experiment. The fusion used on two image situations: *exist*, and *similar*. The algorithms used in the fusion gave better results after applying the fusion.

Chapter 5

Discussion

5.1 Introduction

The main purpose of the proposed system is to assist the visually impaired persons to easily and effectively shop in grocery stores. The only input to the system is represented by images captured from three camera hanging in the shopping cart. Two computer vision algorithms were used to process the captured images for image recognition. The first algorithm is OCR, and it needs only the name of the user desired product as text. The OCR recognized the text written on the products in the shelf view images. The OCR read the text of two imaging conditions: Marketing images and Planogram Front images (PF). The system matched the name of the desired product with the text retrieved from the OCR twice: once with the BN and PD, and once with the PD alone.

The second algorithm was IPM, and it is required images as the user desired product images to be matched with the montage image. For that reason, a database constructed containing 50 images as the user desired product to be matched with the montage image. The IPM performed the matching using two matching strategies: NN and NNDR. The IPM algorithm was considered to be resource consuming because it required building a database of the grocery store products.

In the first phase, we experimented each algorithm alone. Then, we fused the IPM with the OCR to assess if their combination is more efficient.

The number of montage images used as a shelve view images was 100, and it contained 775 products. The system divided the 100 montage images into four situations that can occur in the system. The four situations are *exist*, *not exist*, *similar*, and *twice*. As presented in Chapter four, the results of the confusion matrix, performance rates, the percentage of accuracy, and decision fusion were calculated.

The discussion chapter is organized as follow: section (5.2) discusses the results of the confusion matrix, section (5.3) discusses the results of the performance rates, section (5.4) discusses the results of the percentage of accuracy, section (5.5) discusses the decision fusion, and section (5.6) is the chapter conclusion. The results, according to each situation for each algorithm, were presented in Chapter (4), but for clarification, we repeated them in this chapter.

5.2 Confusion Matrix

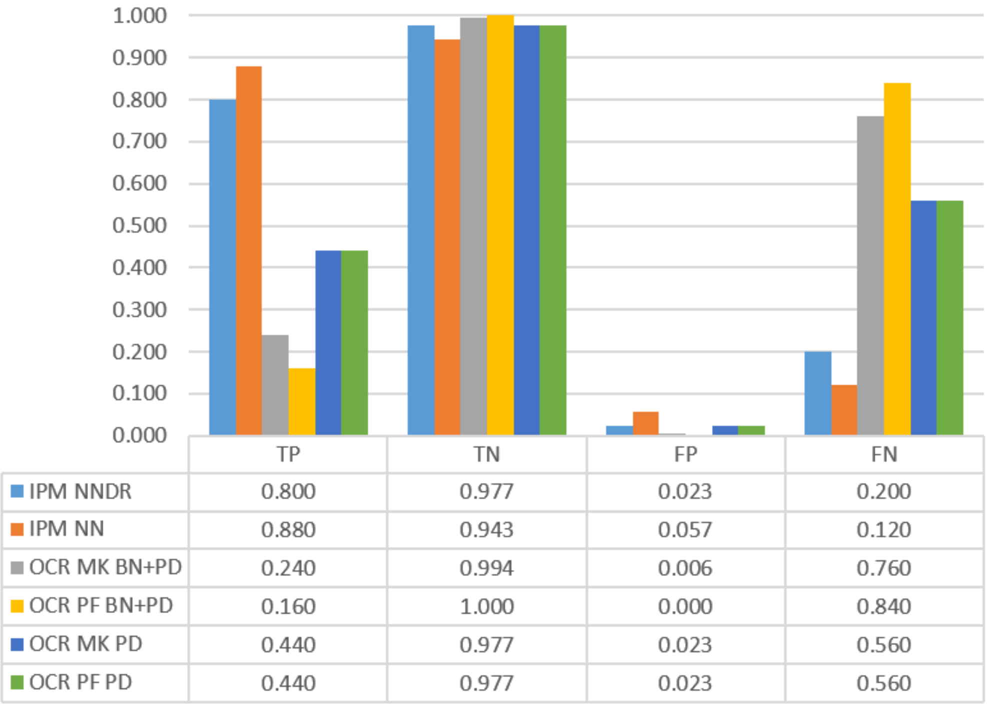
The confusion matrix consist of four rates: TP, TN, FP, and FN, where TP is the number of the truly matched products, TN is the number of the truly non-matched products, FP is the number of the falsely matched products, FN is the number of the falsely non-matched products.

The sum of the TP and FN is the value of all matching products (positive). Therefore, the value of FN completes the value of TP. According to Szeliski in [15], the value of TP should be close to one and the value of FP should be close to zero. The sum of TN and FP is the value of all non-matching products (Negative). Therefore, the value of FP completes the value of TN.

The results are sometimes equal to one or equal to zero and, this is due to the unreliability of OCR when using brand name matching. It is hard for OCR to read the brand names of the products because there are multiple variations in the font, text color, text background and font orientation. Furthermore, OCR may return empty results that do not contain text, and the system considers it as TN in all situations. The empty text results may also cause FN in *exist* and *twice* situations. For example, if the user requested to search for a specific product and the OCR could not recognize it returning an empty text, the system considers it as FN. For that reason, we did not discuss the results of OCR matching using BN and PD.

We used two different matching strategies in the IPM algorithm. The system applied the NN strategy if i) the descriptor of the user desired product image is the nearest neighbor to the descriptor of the montage image; and ii) if the distance between the two descriptors is under the threshold. The NNDR strategy is similar to NN, but the threshold was applied to the distance ratio between the two nearest neighbors. Therefore, the IPM with the NN strategy brings more results than using the IPM with the NNDR. The returned results can be either true match or false match. The reason behind that, the NN strategy apply the threshold directly on the distance between the descriptors while the NNDR strategy apply the threshold to the distance ratio between the descriptors.

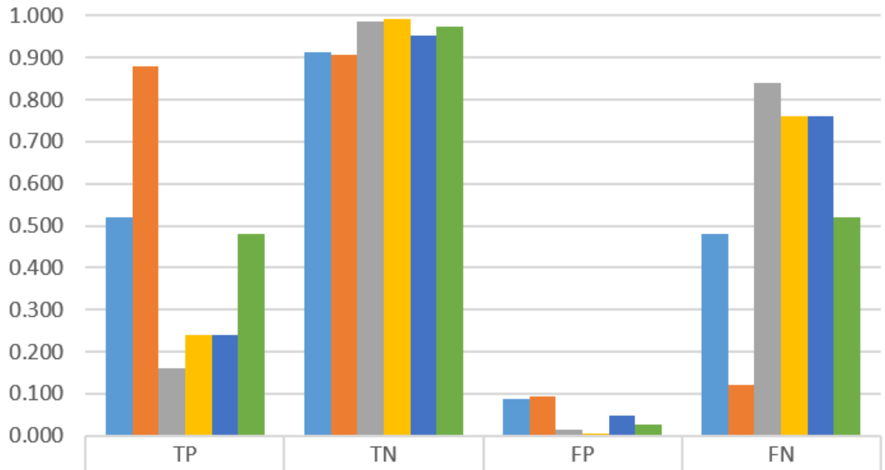
Figure (5.1) shows the results of the confusion matrix rates for *exist* situation, which is when the TP =1 and TN =7. Therefore, the total TNs is higher than the TPs. The IPM NNDR and IPM NN gained the first higher confusion matrix results. The IPM NN gives higher TP and FP result than IPM NNDR because the NN strategy matched with all the nearest neighbors who were under the threshold. On the other hand, the IPM NNDR gave higher TN and FN results than IPM NN. When matching the OCR using PD alone, the results of the MK images are the same of the PF images. The OCR PD matching gave a reasonable TP and FP results as the second higher results.



	TP	TN	FP	FN
🟦 IPM NNDR	0.800	0.977	0.023	0.200
🟧 IPM NN	0.880	0.943	0.057	0.120
⬜ OCR MK BN+PD	0.240	0.994	0.006	0.760
🟨 OCR PF BN+PD	0.160	1.000	0.000	0.840
🟦 OCR MK PD	0.440	0.977	0.023	0.560
🟩 OCR PF PD	0.440	0.977	0.023	0.560

Figure 5.1: Confusion matrix rates for *exist* situation

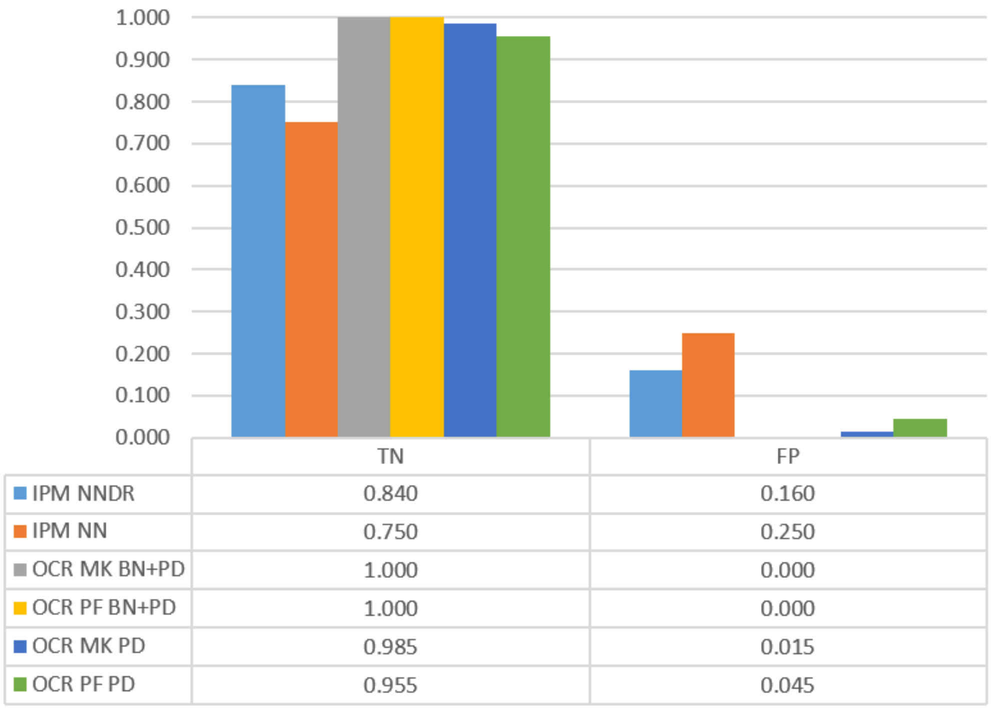
Figure (5.2) shows the results of the confusion matrix rates for a *similar* situation. As discussed in section (4.2), the accurate *twice* situation is when TP =2 and TN =6. Therefore, the total TNs will be higher than the TPs. The IPM NNDR and IPM NN gained the first higher results. The IPM NN gave higher TP and FP result than IPM NNDR because the NN strategy matched with all the nearest neighbors who are under the threshold. On the other hand, the IPM NNDR gave higher TN and FN results than IPM NN. When matching the OCR using PD alone, the results of the TP and TN in PF images is higher than the MK images. The OCR matching using PD in the MK images resulted in a high number of FP and FN comparable with the PF images. The reason behind it is that the PF images represent only the front face of the products, and the text is clear than the text in the MK images as shown in Figure (3.12) and (3.13).



	TP	TN	FP	FN
🟦 IPM NNDR	0.520	0.913	0.087	0.480
🟧 IPM NN	0.880	0.907	0.093	0.120
⬜ OCR MK BN+PD	0.160	0.987	0.013	0.840
🟨 OCR PF BN+PD	0.240	0.993	0.007	0.760
🟦 OCR MK PD	0.240	0.953	0.047	0.760
🟩 OCR PF PD	0.480	0.973	0.027	0.520

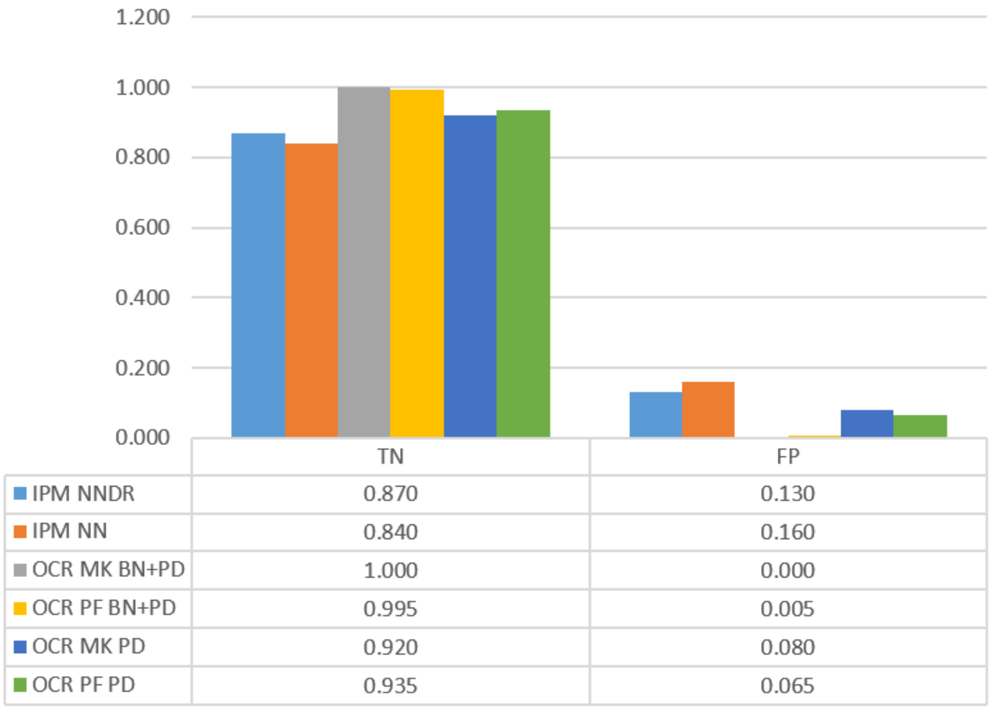
Figure 5.2: Confusion matrix rates for *twice* situation

The *not exist* and *similar* situation means that, the desired product does not exist in the montage images. Therefore, the TP and FN rates are not available and only TN and FP rates were measured. The sum of the TN and FP should equal to eight products because the desired product does not exist in the montage image.



	TN	FP
■ IPM NNDR	0.840	0.160
■ IPM NN	0.750	0.250
■ OCR MK BN+PD	1.000	0.000
■ OCR PF BN+PD	1.000	0.000
■ OCR MK PD	0.985	0.015
■ OCR PF PD	0.955	0.045

Figure 5.3: Confusion matrix rates for *not exist* situation



	TN	FP
■ IPM NNDR	0.870	0.130
■ IPM NN	0.840	0.160
■ OCR MK BN+PD	1.000	0.000
■ OCR PF BN+PD	0.995	0.005
■ OCR MK PD	0.920	0.080
■ OCR PF PD	0.935	0.065

Figure 5.4: Confusion matrix rates for *similar* situation

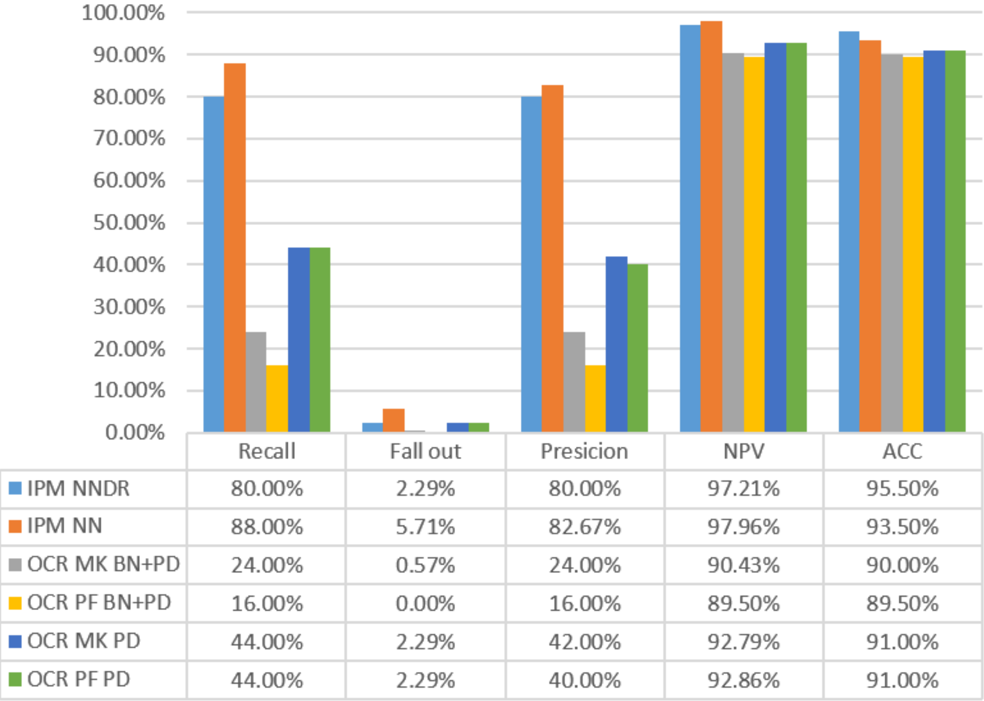
As shown in the Figure (5.3), The IPM using NNDR strategy and OCR PD matching in MK images are the best two algorithms because they gained a higher TN and a lower FP. The IPM using the NN strategy yields to low TN and a high FP because the desired product image interest points could not match with any product in the montage images.

As shown in the Figure (5.4), IPM using NNDR strategy and OCR PD matching in PF images are the best two algorithms because they gained a higher TN and a lower FP. IPM using NN strategy yields to low TN and a high FP because the desired product image interest points could not match with any product in the montage images. The reason behind the high TN and low FP in the OCR PD matching in the PF images is that the OCR can recognize text more easily in the PF images than the MK images. Accordingly, the system is capable to separate the desired product description and the similar product description in the montage image.

5.3 Performance Rate

The performance rates include the Recall (the possibility that a matching product is retrieved), Precision (the possibility that the retrieved product match), Fall out (total number of false products retrieved from the system), NPV (the total of non-match products retrieved), and Accuracy (the system ability to recognize products or to exclude products correctly from images).

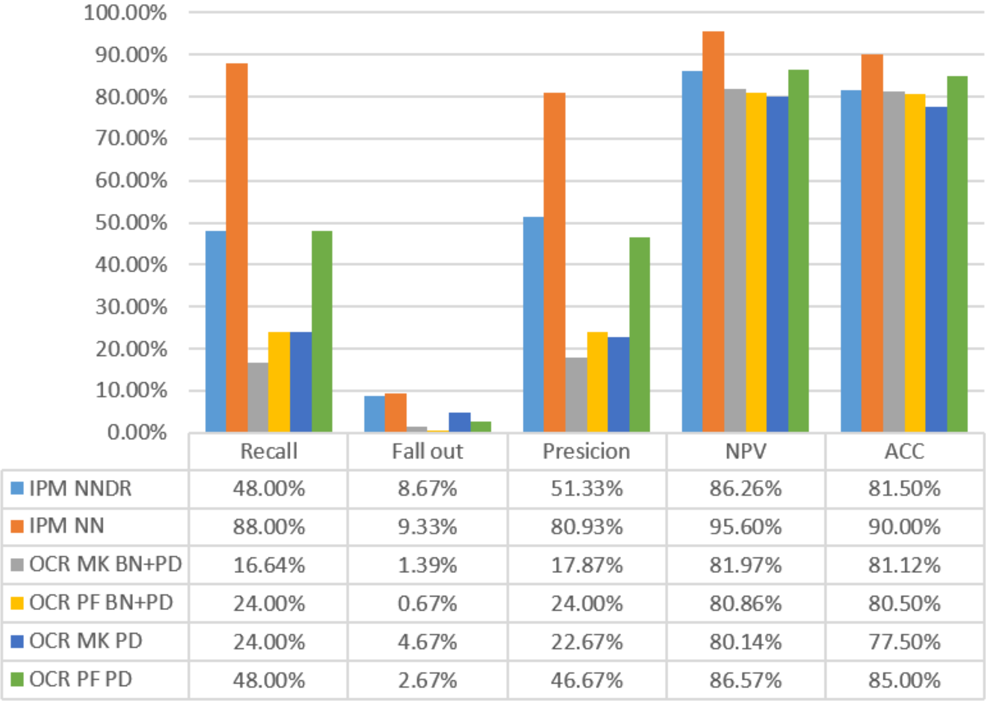
Figure (5.5) shows the performance rates for *exist* situation. The IPM using the NN strategy resulted in a higher Recall, Precision, Fall out, and NPV, but the higher accuracy algorithm was IPM using the NNDR strategy because the number of the fall out that indicates the number of the false positives for IPM NN was low comparable with the other algorithms.



	Recall	Fall out	Presicion	NPV	ACC
🟦 IPM NNDR	80.00%	2.29%	80.00%	97.21%	95.50%
🟧 IPM NN	88.00%	5.71%	82.67%	97.96%	93.50%
⬜ OCR MK BN+PD	24.00%	0.57%	24.00%	90.43%	90.00%
🟨 OCR PF BN+PD	16.00%	0.00%	16.00%	89.50%	89.50%
🟦 OCR MK PD	44.00%	2.29%	42.00%	92.79%	91.00%
🟩 OCR PF PD	44.00%	2.29%	40.00%	92.86%	91.00%

Figure 5.5: Performance rates for *exist* situation

Figure (5.6) shows the performance rates for the twice situation. The most efficient is IPM using NN strategy because the desired product is mentioned two times in the image and the NN strategy matched the montage image with the nearest neighbors of the desired product image that is under the threshold. Thus, the probability of finding the desired product is higher than in the *exist* situation.



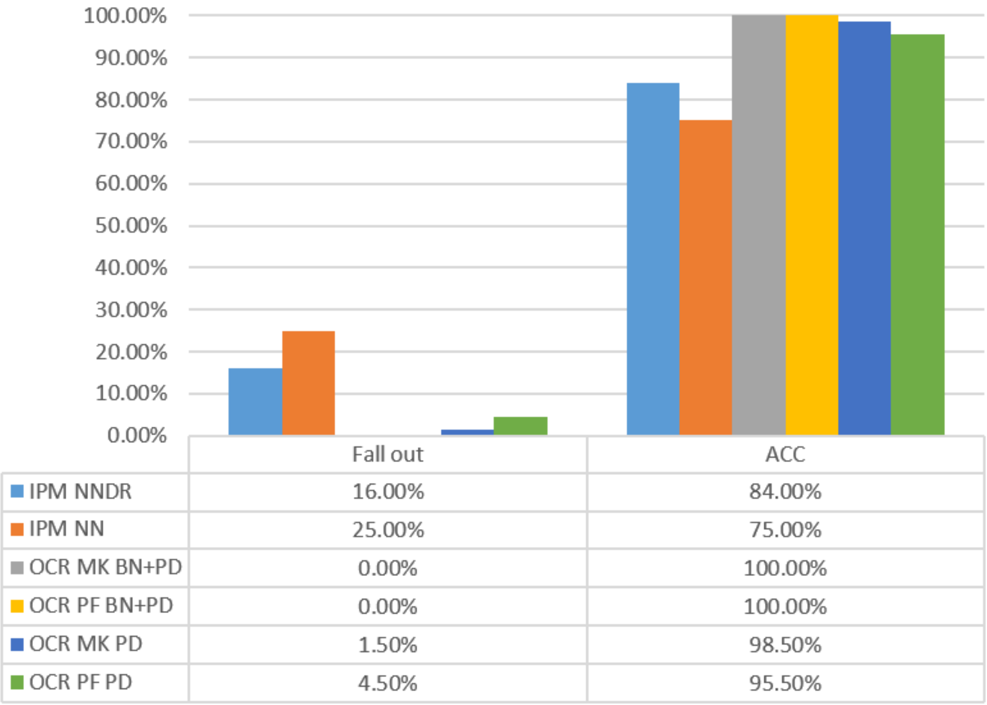
	Recall	Fall out	Presicion	NPV	ACC
🟦 IPM NNDR	48.00%	8.67%	51.33%	86.26%	81.50%
🟧 IPM NN	88.00%	9.33%	80.93%	95.60%	90.00%
⬜ OCR MK BN+PD	16.64%	1.39%	17.87%	81.97%	81.12%
🟨 OCR PF BN+PD	24.00%	0.67%	24.00%	80.86%	80.50%
🟦 OCR MK PD	24.00%	4.67%	22.67%	80.14%	77.50%
🟩 OCR PF PD	48.00%	2.67%	46.67%	86.57%	85.00%

Figure 5.6: Performance rates for *twice* situation

In the *not exist* and *similar* situations, the rates of TP and FN are not available. Therefore, the precision, recall, and NPV cannot be calculated.

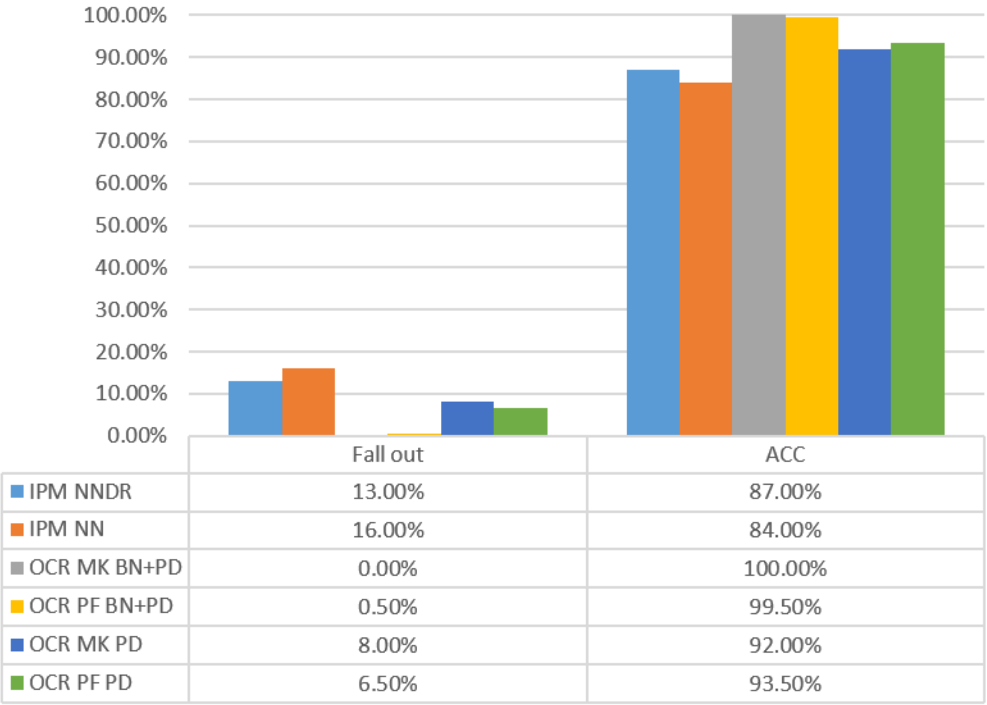
Figure (5.7) shows the performance rate for not exist situation. OCR matching using PD in MK images gained the higher accuracy and lowered fall out results. IPM using the NN strategy resulted in lowest accuracy results and the highest fall out results. The reason behind it is that IPM using NN search for the nearest neighbor's similar product to the desired product under the threshold. Therefore, IPM NN returned the most similar non-matching products to the desired product.

Figure (5.8) shows the performance rate for *similar* situations. The OCR matching using PD in PF images resulted in higher accuracy and lowered fall out results. The text written on the PF product images is clearer than the MK product images and therefore, the OCR can return more accurate results and the OCR can easily distinguish the PD of the user desired product.



	Fall out	ACC
■ IPM NNDR	16.00%	84.00%
■ IPM NN	25.00%	75.00%
■ OCR MK BN+PD	0.00%	100.00%
■ OCR PF BN+PD	0.00%	100.00%
■ OCR MK PD	1.50%	98.50%
■ OCR PF PD	4.50%	95.50%

Figure 5.7: Performance rates for *not exist* situation



	Fall out	ACC
■ IPM NNDR	13.00%	87.00%
■ IPM NN	16.00%	84.00%
■ OCR MK BN+PD	0.00%	100.00%
■ OCR PF BN+PD	0.50%	99.50%
■ OCR MK PD	8.00%	92.00%
■ OCR PF PD	6.50%	93.50%

Figure 5.8: Performance rate for *similar* situation

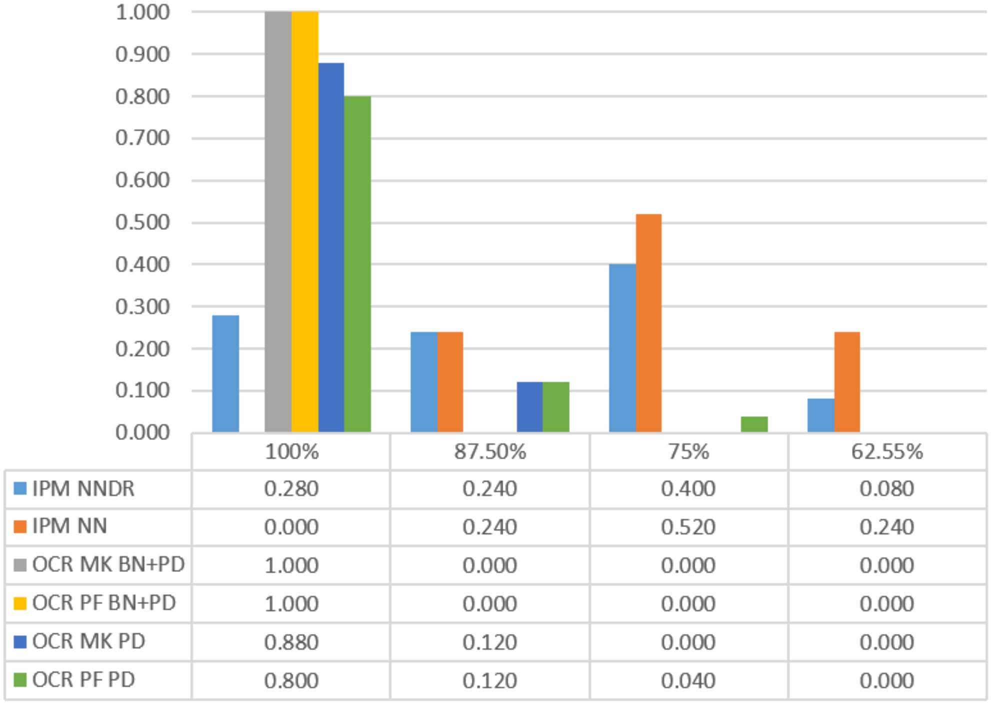
5.4 Percentage of accuracy

According to the unavailability of the precision and recall rates in *not exist* and *similar* situations, we calculated the percentage of accuracy of each situation. The percentage of accuracy calculates the ability of the system to measure each situation accurately 100 % or beyond.

In the *similar* and *not exist* situations, the rate of TN +FP should equal to eight. If the system detected TN = 8 and FP = 0, then the system is 100% accurate. If TN = 7 and FP = 1, then the system is 87.5% accurate. If TN = 6 and FP = 2, then the system is 75% accurate. If TN = 5 and FP = 3, then the system is 62.55% accurate. If TN = 4 and FP =5, then the system is 50% accurate.

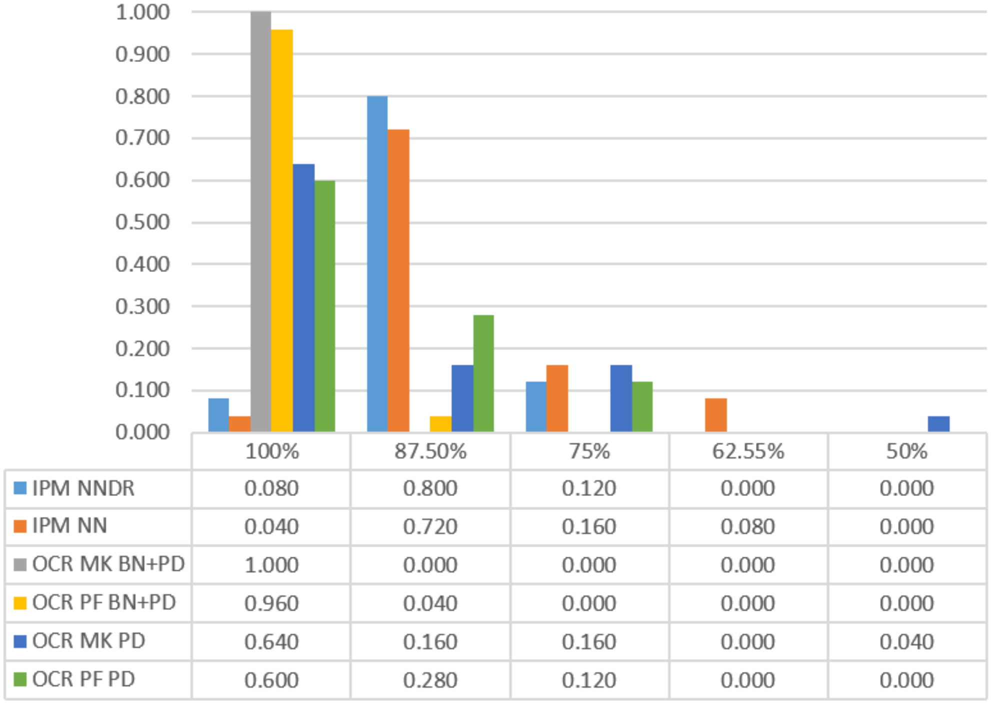
Figure (5.9) shows the percentage of accuracy for *not exist* situation. The OCR matching using BN and PD was eliminated from the discussion due to their unreliability to provide good results. The most accurate algorithm is OCR matching using PD because the number of montage images that matched TN = 8 and FP = 0 is more than in the IPM algorithms. IPM is less accurate because if a matching product was not found, it searches for the neighboring matching products and declare the false neighbors matches as correct matches.

Figure (5.10) shows the percentage of accuracy for a similar situation. The OCR matching using BN and PD was eliminated from the discussion. The OCR matching using PD is the leading algorithms in accuracy. OCR is considered more accurate than IPM because it matches the desired product name as text with the text written on the products under a specific threshold. On the other hand, IPM matches the user desired product image interest points with the shelf view image interest points under a specific threshold. OCR matches the text on the desired product image with text on the products on the montage image. So, that there is no possibility for a similar product to match the desired product because we are matching text by text. While, in IPM the interest points in the desired product image may match number of interest points in the montage image by that there is a door for similar products to match the desired products.



	100%	87.50%	75%	62.55%
IPM NNDR	0.280	0.240	0.400	0.080
IPM NN	0.000	0.240	0.520	0.240
OCR MK BN+PD	1.000	0.000	0.000	0.000
OCR PF BN+PD	1.000	0.000	0.000	0.000
OCR MK PD	0.880	0.120	0.000	0.000
OCR PF PD	0.800	0.120	0.040	0.000

Figure 5.9: Accuracy percentage of *not exist* situation



	100%	87.50%	75%	62.55%	50%
IPM NNDR	0.080	0.800	0.120	0.000	0.000
IPM NN	0.040	0.720	0.160	0.080	0.000
OCR MK BN+PD	1.000	0.000	0.000	0.000	0.000
OCR PF BN+PD	0.960	0.040	0.000	0.000	0.000
OCR MK PD	0.640	0.160	0.160	0.000	0.040
OCR PF PD	0.600	0.280	0.120	0.000	0.000

Figure 5.10: Accuracy percentage for *similar* situation

5.5 Decision Fusion

As described in section (4.5), the Decision fusion was implemented to create a multimodal system constructing from the six algorithm methods used in the system. The two situations that used the fusion are: *exist* and *twice*. In *exist* situation, the fusion was made between IPM NNDR, OCR PF PD, and OCR MK PD. In *twice* situation, the fusion was made between IPM NN, OCR PF PD, and OCR MK PD.

Figure (5.11) shows the fusion results for exist situation. The results of the fusion between the three algorithms are 8 % better than the higher result algorithm (IPM NNDR). Figure (5.12) shows the fusion results for the twice situation. The results of the fusion between the three algorithms are 4 % better than the higher result algorithm (IPM NN).

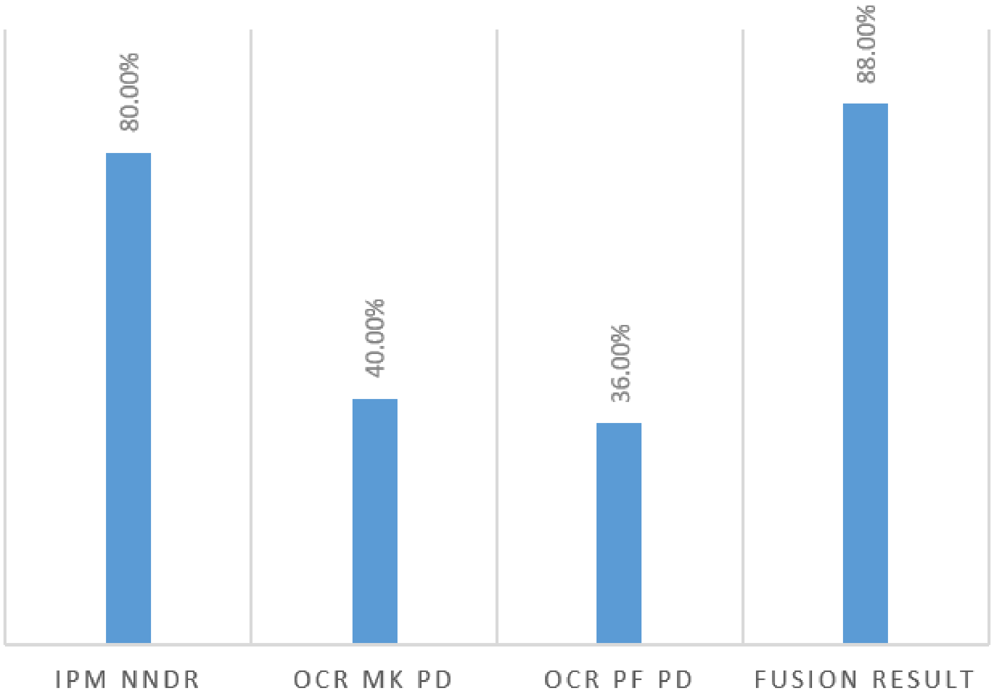


Figure 5.11: Fusion Results for *exist* situation

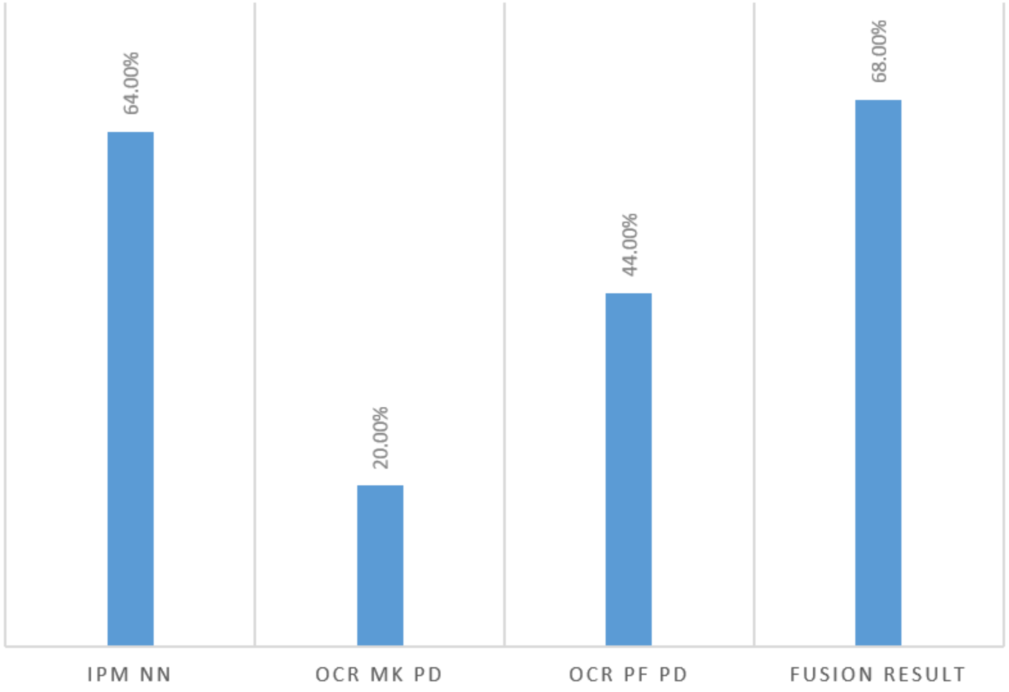


Figure 5.12: Fusion Results for *twice* situation

5.6 Background Systems Results Comparison

We compared the systems mentioned in the Literature Review in section (2.2) with the proposed system. We used the results of *exist* situation for comparison because resemble the results of the other systems. The comparison included: image features matching algorithms, OCR engines, and image features fusion with OCR recognition.

5.6.1 Image Feature Matching Algorithms

As mentioned in section (2.4), there are different image features that can be used for image recognition. Interest point features was used to match between different images using interest point descriptors. The two most known interest point descriptors are: SIFT and SURF. SIFT was used by GroZi (2.3.1.4), who is an object localization and recognition system. It localizes grocery products in image frames captured from the real grocery store. The grocery products were collected from a database called GroZi-120. The SIFT algorithm was evaluated in the system using the precision and recall rates.

Our system used the SURF algorithm for interest point matching (IPM) phase. IPM was tested using two matching strategies mentioned earlier: NN and NNDR. We compared the precision and recall results of the GroZi system using SIFT with our system using SURF algorithm as shown in Table (5.1). The table shows that the SURF algorithm is better than SIFT algorithm in precision and recall rate. The SIFT has a low precision rate because of the high number of the FP as stated in [28].

Table 5.1: Recall and Precision comparison between GroZi and our system

Systems		Recall	Precision
GroZi using SIFT Algorithm		0.72	0.18
Our System using SURF Algorithm	NN Strategy	0.88	0.82
	NNDR Strategy	0.80	0.80

5.6.2 OCR Engines

There are two most famous OCR engines: Tesseract (TESS) and ABBYY Fine Reader. A comparison was made between TESS and ABBYY engines in [39] according to the precision and recall rates on the ICDAR 2003 dataset. We will compare the results of the TESS and ABBYY in [39] and our system results using ABBYY, as shown in Table (5.2). ABBYY was implemented on two image criteria: MK and PF. Furthermore, the ABBYY results in our system were matched according to two methods: matching BN and PD, and matching PD alone. Matching with the BN and PD was eliminated from the comparison due to the hardness of BN recognition.

Table (5.2) shows that our implementation of ABBYY engine resulted in better recall rates, while the ABBYY implementation in [39] using ICDAR dataset performed better precision rates. The ICDAR dataset is a dataset containing images captured from the street and not specialized for grocery store products. Furthermore, ICDAR contains training images to train the algorithms before testing.

Table 5.2: OCR engines comparison between [39]and our system

System		**Recall**	**Precision**
TESS [39]		0.18	0.35
ABBYY [39]		0.32	0.71
Our System using ABBYY (matching with PD)	**MK images**	0.44	0.42
	PF images	0.44	0.40

5.6.3 Image Features fusion with Text Recognition

As mentioned in section (2.6.2), a multimodal system was created [39] using the fusion between the image features and text recognition on IMET dataset. The system developed a new visual saliency cues for detecting text regions in images. The fused image features algorithm was SIFT using Bag of Visual Words scheme (BOW). It applied two text recognition algorithms, the system either using the visual saliency cues output direct to OCR (ocr), or using the visual saliency cues in the ground truth bounding boxes around the text in the image (ocr_bb). The ground truth bounding boxes means the exact framing of a specific object in the scene image. The multimodal system measured the accuracy of the images features fused with the two text recognition algorithms. Table (5.3) shows the comparison between the accuracy results of the multimodal fusion results and our multimodal fusion results (in both *exist* and *twice* situations).

Table 5.3: Fusion results comparison between [39] and our multimodal system

Multimodal Systems	BOW + ocr [39]	BOW + ocr_bb [39]	Our Multimodal System	
			exist situation	*twice* situation
Accuracy	0.816	0.839	0.98	0.90

Table (5.3) shows that our multimodal system results in better accuracy rates than the multimodal of [39]. They used the SIFT algorithm for the fused image features which were proved that it is less accurate than the SURF algorithm used in our multimodal system.

5.7 Conclusion

The chapter discussed the results of the confusion matrix, performance rate, the percentage of accuracy, and the results of the decision fusion. Furthermore, a comparison between each feature used in our system with other systems that used the same features was introduced. Additionally, our fusion results were compared with other fusion results used similar image and text features.

The confusion matrix results showed that the IPM is better than the OCR when the system computes the positive results (TP and FN) in *exist* and *twice* situations. On the other hand, OCR is better for measuring the negative results (TN and FP) in the four situations.

In case the desired product is available on the shelf, the performance rates showed that the IPM is more accurate than the OCR because IPM is capable to bring more either matching or not results. IPM using NNDR strategy is more accurate than NN *in exist* situation because the number of FP in NNDR strategy is less than in NN strategy. On the other hand, in *not exist* and *similar* situations, OCR algorithm is better than the IPM because it is capable to accurately distinguish between the TN and FP.

The percentage of accuracy results showed that OCR algorithm is better than IPM in case the desired product does not exist in the shelf view images. The reason behind it is that IPM brings more positive products that may not match in real life the desired product. The decision fusion results showed a reasonable enhancement after applying the fusion between OCR and IPM.

The comparisons between our system and the background systems showed that our system performed better in precision, recall, and accuracy results. The ABBYY OCR engine depends on the input dataset images and the preprocessing and training phases before the recognition phase.

Chapter 6

Conclusion

The visually impaired persons require assistance for daily actions such as grocery shopping. The current thesis aims to develop a system that can assist the visually impaired at grocery shopping.

Chapter one is a brief introduction to the thesis. Chapter two stated the systems developed for assisting the visually impaired persons to individually perform grocery shopping. Some systems navigate the user inside the supermarket while others assist in locating the required product. We compared the advantages, disadvantages, algorithms, and features of the background systems. The common problem in the background systems is portability such that many systems overload the user with equipment's or operations. Additionally, the wireless and database techniques may prevent the systems from being used in any grocery store.

In order to recognize objects within an image, there are different algorithms that can be applied. Object recognition can be implemented using different features in the image such as visual features or text features. Visual features can be used to match between images for recognition such as Interest Point matching (IPM), while text features can be recognized using Optical Character Recognition (OCR). The most famous IPM algorithms are SIFT and SURF. The OCR results can be obtained using OCR engines such as ABBYY and Tesseract. In order to evaluate the matching results, the confusion matrix and performance rates concepts are introduced. Each rate is defined with the corresponding equation.

According to the problem raised in the background systems, it is important to develop a system that does not require a lot of portable equipment and that minimizes the system operating missions for the user. Additionally, the system should operate in any grocery store without restrictions to a wireless connection or database to recognize products, but uses computer vision techniques for object recognition. Therefore, in the thesis we created a system consisting of a shopping cart with three cameras installed vertically on one side of the cart.

Chapter three summarized the system workflow into three main stages: i) announcing the aisle category name to the user, ii) finding the user desired product on the shelf, iii) guiding the user to the product location. In order to find the desired product, we tested the SURF algorithm and ABBYY OCR engine. The SURF algorithm is applied with two different matching strategies. The test is conducted on dataset rather than taking a real images from the grocery store.

The algorithms results are evaluated in chapter four according to the confusion matrix rates and performance rates. The calculation procedure used to calculate the confusion matrix and the performance rate is described and outlined in a graph. An example of each result rate is shown to clarify the procedures used. A modified method is used to calculate the accuracy percentage of the results in case the performance rates were not available. We described the procedure used to apply the fusion techniques to fuse between the SURF algorithm results and ABBYY results.

Chapter five discussed the results of the confusion matrix, performance rate, and percentage of accuracy according the minimum and maximum rates. Also, we discussed the unfamiliar results with an explanation for each case. The results showed that the IPM algorithm performed better than the OCR algorithm in case the user desired product is available within aisle shelves. In contrast, OCR algorithm performed better than IPM in case the desired product is not available on the shelves. The reason behind it is that IPM is capable to find more true matching products but at the same time brings many false matching products. On the other hand, OCR cannot find many true matching products but at the same time brings more true non-matching products.

A comparison is made between the results of our system and the background systems that used the same image and text features. The comparison showed that, our system performed better than the other system in image feature recognition, without the need for training data.

Additionally, a comparison between the algorithms results before and after the fusion is discussed. Finally, a comparison was made between the results of the background fusion multimodal and our fusion multimodal. The fusion background systems used different image features and the same text features. The fusion comparison showed that our fusion multimodal approach performed better than the other variant.

We concluded the thesis with the most important findings and the thesis limitations. Finally, based on the thesis findings, some future work recommendations are provided.

6.1 Limitations

The proposed system, although better than the existing variants has a series of limitations and was developed taking into consideration a series of aspects such as:

1- According to the three stage system workflow, the thesis executed only the part of recognizing the desired product using image recognition techniques. The aim was to compare the results of the two methods and decide which one will be applicable to use for the system.

2- The system is limited to be applied only on products described with English language.

3- The shelf view images that assumed to be captured from the grocery store were replaced by montage image built from different product images in the item master dataset.

4- The system used to operate to shopping cart was developed, but the hardware was not constructed. Due to that, we did not discussed the configurations of the shopping cart and the cameras specifications.

5- The system used a product images database for matching between the users desired products images and the shelf view images. This somewhat contradicts the aim of the system to use it in any grocery store, but one must take into consideration that every grocery store contains if not the same product, very similar ones.

6- We are assuming that the products on the shelves are positioned correctly in the front face of the products. Although in the real life not all the products are aligned correctly, but it is also impossible to find that all the products are aligned incorrectly. Furthermore, each product is repeated on the shelf. So, the cameras can detect the ones that is correctly aligned.

6.2 Future Work

In the future, we can develop an OCR algorithm that is specially created for detecting and recognizing text written on grocery store products. Furthermore, we can use a database containing images for the products brand names logo. This can solve the issue of the hard recognition of products brand names due to their variety in the font, color, and orientation. Additionally, we can use the grocery store categories as input lexicon for character recognition.

The products images that are captured from the cameras can be processed to detect each product alone in ground truth box separately. That can benefit the system by detecting the most important text areas in the product image such as the brand name and product description.

The system can be improved by including a navigation system to direct the user inside the grocery store, cashier, and entrance. The navigational system can use the signs inside the grocery store (aisle category name sign, cashier sign, entrance sign) to direct the user.

LIST OF REFERENCES

[1] World Health Organization, "Visual impairment and blindness," Augest 2014. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs282/en/. [Accessed 5 May 2015].

[2] V. Kulyukin et al., "RoboCart: Toward Robot-Assisted Navigation of Grocery Stores by the Visually Impaired," in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, Alberta, 2005.

[3] C. P. Gharpure and V. A. Kulyukin, "Robort-assisted shopping for the blind: issues in spatial cognition and product selection," *Intelligent Service Robotics*, pp. 237-251, 2008.

[4] J. Nicholson et al., "ShopTalk: Independent Blind Shopping Through Verbal Route Directions and Barcode Scans," *The Open Rehabilitation Journal,* vol. 2, 2009.

[5] V. Kulyukin and A. Kutiyanawala, "Demo: ShopMobile II: Eyes-Free Supermarket Grocery Shopping for Visually Impaired Mobile Phone Users," in *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010.

[6] P. Lanigan et al., "Trinetra: Assistive Technology for Grocery Shopping for the Blind," in *10th IEEE International Symposium on Wearable Computers*, Montreux, 2006.

[7] P. Narasimhan, "Assistive Embedded Technologies," *IEEE Computer Magazine*, vol. 39, pp. 85-87, 2006.

[8] L. Carlson et al., "GroZi Shopping Assistant," California Institute for Telecommunications and Information Technology, San Diego, 2007.

[9] R. Tran et al., "GroZi Multitouch Tablet Prototype," 2011. [Online]. Available: http://grozi.calit2.net/files/TIESGroZiFa11.pdf. [Accessed 18 January 2015].

[10] R. Tran and C. Taira, "GroZi Android Tablet," 2012. [Online]. Available: http://grozi.calit2.net/files/TIESGroZiWi12.pdf. [Accessed 18 January 2015].

[11] S. Krishna et al., "A Wearable Wireless RFID System for Accessible Shopping Environments," in *3th International Conference on Body Area Networks (BodyNets08)*, Arizona, 2008.

[12] D. Ipina et al., "BlindShopping: Enabling Accessible Shopping for Visually Impaired People through Mobile Technologies," in *9th International Conference on Smart Homes and Health Telematics*, Montreal, 2011.

[13] R. Fisher, et al., Dictionary of Computer Vision and Image Processing, Somerset, NJ: John Wiley & Sons, 2013.

[14] M. Treiber, "An Introduction to Object Recognition," in *Advances in Pattern Recognition*, London, UK: Springer-Verlag, 2010.

[15] R. Szeliski, Computer Vision Algorithms and Applications, London, UK: Springer-Verlag, 2011.

[16] A. Baumberg, "Reliable Feature Matching Across Widely Separated Views," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2000.

[17] A. Yadav and P. Yadav, Digital Image Processing, New Delhi, India: Laxmi, 2009.

[18] A. Goshtasby, 2D and 3D Image Registration for Medical, Remote Sensing, and Industrial Applications, New Jersey: John Willy & Sons, 2005.

[19] C. Harris and M. Stephens, "A combined corner and edge detector," in *4th Alvey Vision Conference*, 1988.

[20] T. Lindeberg, "Feature detection with automatic scale selection," *IJCV*, vol. 30, no. 2, p. 79 – 116, 1998.

[21] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," in *ICCV*, 2001.

[22] D. Lowe, "Object recognition from local scale-invariant features," in *ICCV*, 1999.

[23] H. Bay, et al, "SURF: Speeded Up Robust Features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, p. 346–359, 2008.

[24] M. Z. e. al., "Evaluation of Interest Point Detectors for Scenes with Challenging Lightening Conditions," in *34th International Conference on Telecommunications and Signal Processing (TSP)*, 2011.

[25] D. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV,* vol. 60, no. 2, p. 91 – 110, 2004.

[26] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *CVPR (2)*, 2004.

[27] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transaction on Pattern Analysis and Maching Intelligence,* vol. 27, no. 10, p. 1615–1630, 2005.

[28] M. Merler et al., "Recognizing Groceries in situ Using in vitro Training Data," in *2nd International Workshop on Semantic Learning Applications in Multimedia (SLAM)*, Minneapolis, 2007.

[29] T. Winlock et al., "Toward Real-Time Grocery Detection for the Visually Impaired," in *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010.

[30] K. Wang and S. Belongie, "Word Spotting in the Wild," in *11th European Conference on Computer Vision*, Greece, 2010.

[31] "ABBYY SDKs for Developers," ABBYY, 2015. [Online]. Available: http://www.abbyy-developers.eu/. [Accessed 20 January 2015].

[32] S. Rice et al., "The Fourth Annual Test of OCR Accuracy, Technical Report 95-03," Information Science Research Institute, Las Vegas, 1995.

[33] R. Smith, "An Overview of the Tesseract OCR Engine," in *9th International Conference on Document Analysis and Recognition (ICDAR)*, 2007.

[34] R. Smith, "Tesseract OCR Engine," OSCON, 2007. [Online]. Available: https://tesseract-ocr.googlecode.com/files/TesseractOSCON.pdf. [Accessed 23 January 2015].

[35] K. Wang et al., "End-to-End Scene Text Recognition," in *IEEE International Coference on Computer Vision (ICCV)*, Barcelona, 2011.

[36] T. Dunstone and N. Yager, "Multimodal Systems," in *Biometric System and Data Analysis*, New York, Springer, 2009, pp. 71-80.

[37] M. Derawi, "Research on Different Biometric Modalities," 2007. [Online]. Available: http://www.biometrics.derawi.com/?page_id=101. [Accessed 8 February 2015].

[38] A. Ross et al., "Levels of Fusion in Biometrics," in *Handbook of Multibiometrics*, New York, Springer, 2006, pp. 73-83.

[39] S. Karaoglu et al., "Object Reading: Text Recognition for Object Recognition," in *Computer Vision – ECCV 2012. Workshops and Demonstrations*, vol. 7585, Florence, Italy: Springer Berlin Heidelberg, 2012, pp. 456-465.

[40] "Shutterstock," 2013. [Online]. Available: http://www.shutterstock.com. [Accessed 31 December 2014].

[41] "ItemMaster," 2012. [Online]. Available: http://www.itemmaster.com. [Accessed 31 December 2014].

أظهرت أداء أفضل من خوارزمية (IPM) في حالة المنتج المطلوب غير متوفر على رفوف الممر. والسبب وراء ذلك، هو أن (IPM) قادرة على العثور على العديد من المنتجات المطابقة في الحقيقية ولكن في نفس الوقت يجلب العديد من المنتجات الخاطئه في المطابقة. في حين، (OCR) لا يستطيع العثور على العديد من المنتجات المطابقة في الحقيقية ولكن في نفس الوقت يجلب العديد من المنتجات الغير المطابقة في الحقيقية.

تم إجراء مقارنة بين نتائج نظامنا والنظم المكورة مسبقاً و التي تستخدم نفس مميزات الصورة والنص. وأظهرت المقارنة أن نظامنا أعطى أداء أفضل من النظم الأخرى في التعرف على مميزات الصور والنص دون الحاجة المسبقة لتدريب النظام كما هو مستخدم في الأنظمة المسبقة الذكر.

بالإضافة إلى ذلك، ناقشنا المقارنة بين نتائج الأندماج قبل وبعد. بعد تطبيق الأندماج أظهرت النتائج زيادة بنسبة 8٪. تم إجراء مقارنة بين نتائج الأندماج للأمثلة الأنظمة متعددة الوسائط و الأندماج الذي تم تطويرة في الأطروحة. استخدمت الأنظمة السابقة للأندماج مميزات للصورة مختلفة ومميزات للنص مماثلة لما تم أستخدامه في الأطروحة. وأظهرت المقارنة أن الأندماج المستخدم في الأطروحة أعطى أداء أفضل من الاندماج المسبق أستخدامة.و قد تم أستنتاج أهم النتائج والقيود في الأطروحة. وأخيراً، استناداً إلى نتائج الأطروحة فقد قمنا بوضع خطط مستقبليه يمكن تطبيقها لتحسين النتائج.

خوارزميات التعرف على العناصر. وأيضاً، تم مقارنة نتائج الخوارزمية (SURF) ومحرك (ABBYY). ثم استخدمنا تقنيات الأندماج لتطوير نظام محتوي على نتائج (SURF) و(ABBYY) .

تتكون بنية النظام المقترح من ثلاث مراحل رئيسية هي: أ) الإعلان عن اسم فئة الممر للمستخدم، ب) العثور على المنتج المطلوب من قبل المستخدم على الرف، ج) توجيه المستخدم إلى موقع المنتج على الرف. من أجل العثور على المنتج الذي طلبة المستخدم، تم أختبار الخوارزمية (SURF) والمحرك (ABBYY). يتم تطبيق الخوارزمية (SURF) بإستخدام استراتيجيتين محتلفتين للمطابقة بين الصور.

وأجريت التجربة على قاعدة بيانات على الويب بدلاً من أخذ صور حقيقية من متجر غذائي. صور المنتجات المستخدمة هي 1550 صورة من اثنين من النماذج شكلية مختلفة. كل تم تعريف كل منتج بإستخدام اسم المنتج الممثل في قبل قاعدة البيانات. سيستخدم اسم المنتج للمطابقة بين المنتج الذي طلبة المستخدم والمنتجات الموجودة على الرفوف في حال المطابقه بإستخدام محرك (OCR). اسم المنتج يحتوي على اسم العلامة التجارية ووصف المنتج. ستتم مطابقة نتائج (OCR) وفقاً لمعيارين مختلفين: المطابقة مع اسم العلامة التجارية للمنتج ووصف المنتج، والمطابقة مع وصف المنتج بمفردة. يتم تصنيف مجموعة البيانات المستخدمة في إطار أربع حالات قد تحدث للمستخدم في المتجر الغذائي عند طلب منتج معين: أ) وجود المنتج المطلوب على رف الممر، ب) عدم وجود المنتج المطلوب على رف الممر، ج) المنتج المطلوب موجود مرتين على رف الممر، د) منتج مماثل للمنتج المطلوب موجود على رف الممر.

سيتم تقييم نتائج الخوارزميات وفقاً لمعادلات مصفوفة المطابقة ومعدلات تقييم الأداء. و لتوضيح الإجراءات المتبعة تم تطبيق أمثله على طريقه الحساب. وفي حال عدم التمكن من تقييم أداء الخوارزميات، تم استخدام طريقة حساب دقة الخوارزميات بالنسبة المئوية. وتم وصف الإجراءات المتبعة لتطبيق تقنيات الدمج بين نتائج خوارزمية (SURF)و(ABBYY). وعلاوة على ذلك، تم استخدام الجداول والأرقام لإظهار نتائج الدمج بين الخوارزميات مقارنة مع نتائج كل خوارزمية بمفردها. تم أيضاح الإجراءات المستخدمة للدمج بإستخدام الأمثلة.

تم مناقشة نتائج مصفوفة المطابقة، ومعدل الأداء، والنسبة المئوية للدقة وفقاً للحد الأدنى والحد الأقصى للقيم. كما ناقشنا النتائج الغير مألوفة مع شرح لكل نتيجة. وأظهرت النتائج أن خوارزمية (IPM) أظهرت أداء أفضل من خوارزمية التعرف الضوئي على الحروف في حالة المنتج المطلوب هو متاح في رفوف الممر. في المقابل، فإن خوارزمية التعرف الضوئي على الحروف

الأنظمة. بالإضافة إلى ذلك، أستخدام التقنيات اللاسلكية وقواعد البيانات يمكن ان تَحُد من تطبيق الأنظمة في أي من المتاجر الغذائية.

من أجل التعرف على العناصر داخل صورة ما، هناك خوارزميات مختلفة لتطبيق مهمه التعرف على العناصر (Object Recognition). يمكن التعرف على العناصر باستخدام مميزات مختلفة في الصورة مثل المميزات بصرية أو المميزات النصية. المميزات بصرية يمكن استخدامها للتعرف على الصور بإستخدام المطابقة بين الصور المختلفة وتسمى بالمطابقة بإستخدام النقط المهمة (Interest Point Matching-IPM). في حين المميزات النصية يمكنها التعرف باستخدام التعرف الضوئي على الحروف (Optical Character Recognition-OCR). من الخوارزميات الأكثر شهرة في المطابقه بإستخدام النقط المهمة (IPM) هي (Scale Invariant Feature Transform-SIFT)و (Speeded Up Robust Features-SURF). يمكن الحصول على نتائج التعرف الضوئي على الحروف (OCR) باستخدام محركات على الويب تقوم بالعملية مثل ABBYY وTesseract.

من أجل تقييم نتائج المطابقة بين المنتج المطلوب من قبل المستخدم و المنتجات المعروضة على رفوف ممرات المتجر الغذائي، قمنا بإستخدام مصفوفات المطابقة (Confusion Matrix) ومعدلات تقييم الأداء(Performance Rates) وتعريفها بإستخدام المعادلات الخاصة بها.

النظام المتعدد الوسائط هو النظام الذي يمكن أن يستفيد من أنظمة مختلفة لتكوين نظام موحد. دمج العديد من الأنظمة المختلفة في نظام واحد يدعى بالأندماج (Fusion). هناك مستويات مختلفة للاندماج وتم شرح الطرق المستخدمة لكل مستوى في الأطروحة. وعلاوة على ذلك، تم النظر الى أحدى تطبيقات الأندماج لإظهار كيف يمكن تنفيذ الأندماج بين المميزات البصرية والمميزات النصية.

وفقاً لمشاكل النظم المذكورة سلفاً في أول الملخص ، فمن الضروري وضع نظام لا يتطلب الكثير من الأجهزة المحمولة أو الطلبات المتعددة لتشغيل النظام من قبل المستخدم. بالإضافة إلى ذلك، يجب أن يعمل النظام في أي متجر غذائي دون الحاجة إلى إستخدام أي اتصال لاسلكي أو قاعدة بيانات للتعرف على المنتجات. بدلاً من ذلك، ينبغي أن نطبق خوارزميات التعرف على العناصر (Object Recognition Algorithms) للتعرف على منتجات المتجر. لذلك، في الأطروحة أنشأنا نظام يتكون من عربة للتسوق مع ثلاث كاميرات مثبتة عمودياً على جانب واحد من العربة. و تم معالجة لقطات الكاميرا تحت

نحو جهاز مساعد للتعرف على أقسام المتاجر الغذائيه والسلع للمصابين بضعف النظر

داليا عصام عطاس

الملخص

يحتاج الأشخاص ذوي الأعاقة البصرية للمساعدة في المهام اليومية مثل التسوق في المتاجر الغذائية. لذلك تهدف الرسالة إلى وضع نظام يمكن أن يساعد الأشخاص ذوي الأعاقة البصرية في التسوق للمتاجر الغذائية.

تعددت النظم التي وضعت لمساعدة الأشخاص ذوي الأعاقة البصرية للتسوق في المتاجر الغذائية بشكل فردي دون الأستعانه بأشخاص آخرين. بعض الأنظمة تساعد المستخدم في التنقل داخل السوبر ماركت في حين أنظمه أخرى تساعد المستخدم في العثور على المنتج المطلوب. أجرينا مقارنة بين المزايا والأيجابيات و السلبيات و الخوارزميات المستخدمه للأنظمه المذكورة. هناك مشاكل شائعة في الأنظمة المذكورة على سبيل المثال تحميل المستخدم بالعديد من المعدات و العمليات المطلوبه لتشغيل

المستخلص

هناك العديد من الأشخاص ذوي الإعاقات البصرية في جميع أنحاء العالم الذين يحتاجون إلى المساعدة في مهام الحياة اليومية مثل التسوق في المتاجرالغذائية. يحتاج الأشخاص ذوي الإعاقات البصرية عادة الى المساعدة من شخص اخر أو من أداة مساعدة في التسوق في المتاجر الغذائيه. يجب أن يبنى نظام للمساعدة في التسوق في المتاجرالغذائية من أجل الحفاظ على خصوصية و استقلالية ذوي الإعاقات البصرية.

هناك العديد من الأنظمة التي تم تصميمها لمساعدة ذوي الإعاقات البصرية للتسوق في المتاجرالغذائية. وهذه الأنظمة التي تم تصميمها تتطلب عملاً ضخماً من المستخدمين لتشغيل أجهزة النظام . وعلاوة على ذلك، فإن الأنظمة تتطلب الاتصالات اللاسلكية و قاعدة بيانات للمنتجات للحصول على معلومات عن المنتجات. هنا تأتي الحاجة لنظام مساعد لذوي الإعاقات البصرية للتسوق في المتاجرالغذائية مع عربة للتسوق دون أي أجهزة إضافية. وفضلاً عن ذلك، ينبغي للنظام استخدام خوارزميات التعرف على العناصر (Object Recognition Algorithms) بدلاً من الاتصالات اللاسلكية وقاعدة البيانات للتعرف على المنتجات.

في هذه الرساله، تم إنشاء نظام لإيجاد حل لمشكلة مساعدة ذوي الإعاقات البصرية للتسوق في المتاجرالغذائية. تتكون بنية النظام من ثلاث مراحل: أ) الإعلان عن أسم فئة الممر للمستخدم، والثاني) العثور على المنتج المطلوب من قبل المستخدم على الرف، والثالث) توجيه المستخدم إلى موقع المنتج. و في هذه الرساله، تم أقتراح تنفيذ عربة للتسوق تتكون من ثلاث كاميرات مثبتة عمودياً على جانب واحد من العربة.

بالإضافة إلى ذلك ، تم مقارنة اثنين من خوارزميات التعرف على العناصر للتعرف على المنتجات الموجودة على رفوف الممر. وكما تمّ أيضاً ، إنشاء نظام متعدد الوسائط لدمج نتائج الخوارزميات المستخدمة للتعرف على العناصر. وأظهرت النتائج أن الدمج أعطى نتائج أفضل من استخدام كل خوارزمية بمفردها.

نحو جهاز مساعد للتعرف على أقسام المتاجر الغذائيه والسلع للمصابين بضعف النظر

داليا عصام عطاس

بحث مقدم لنيل درجه الماجستير في علوم الحاسبات

د/ وديع الحلبي

كلية الحاسبات و تقنيه المعلومات
جامعة الملك عبدالعزيز - جدة
رمضان 1436 هـ - يونيو 2015 م

بسم الله الرحمن الرحيم

قال الله تعالى

{ يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ }

سورة المجادلة آية رقم (١١)