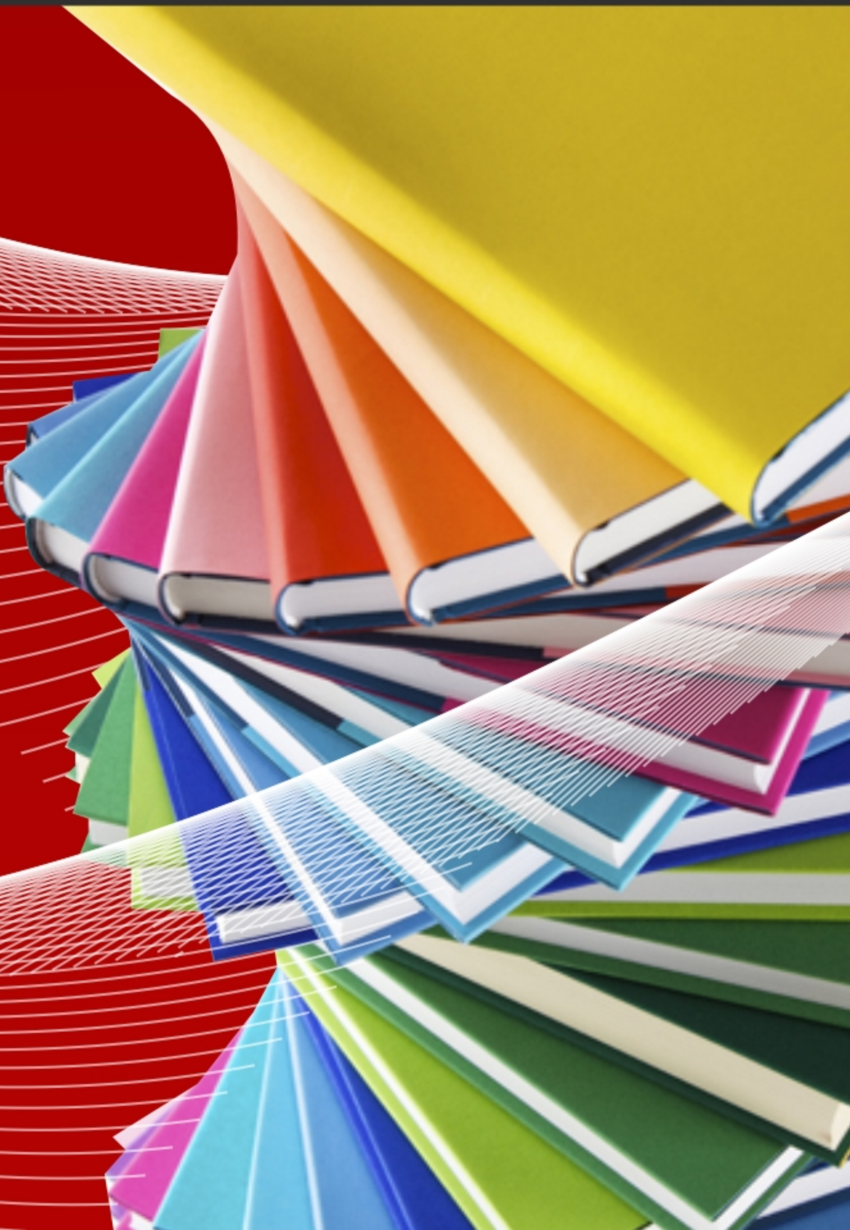
澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Outstanding Academic Papers by Students
學 生 優 秀 作 品



University of Macau

Faculty of Science and Technology

澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Indoor Navigation by Image Recognition

by

Hong Ka Wo, Louis, Student No: D-B3-2518-3

Final Project Report submitted in partial fulfillment
of the requirements of the Degree of
Bachelor of Science in Computer Science

Project Supervisor

Prof. Pun Chi Man

26 May 2017

DECLARATION

I sincerely declare that:

1. I and my teammates are the sole authors of this report,
2. All the information contained in this report is certain and correct to the best of my knowledge,
3. I declare that the thesis here submitted is original except for the source materials explicitly acknowledged and that this thesis or parts of this thesis have not been previously submitted for the same degree or for a different degree, and
4. I also acknowledge that I am aware of the Rules on Handling Student Academic Dishonesty and the Regulations of the Student Discipline of the University of Macau.

Signature : ______________________________

Name : Hong Ka Wo, Louis

Student No. : D-B3-2518-3

Date : 26 May 2017

ACKNOWLEDGEMENTS

First and foremost, we would like to show my utmost gratitude to the Honours College of University of Macau for providing the opportunity to carry out a project as a partial fulfilment of the requirement for the Honours Programme. Throughout this project, we were very fortunate to receive the guidance and encouragement from my supervisor Prof. Pun Chi Man. We are truly grateful for his patience as well as the resources he has provided.

ABSTRACT

Image recognition technology has become increasingly popular in daily life since smartphones are also increasingly powerful. The purpose of smartphone is no longer limited to communication, but also entertainment, computation, etc. Navigation is one of the major usage of smartphone. Global Positioning System (GPS) is widely used in the outdoor environment for navigation. However, GPS signal is strongly influenced in indoor environment and hence not applicable. Many researchers have proposed various alternative positioning methods. For instance, a fusion approach combining Wi-Fi and Dead Reckoning technique, and an approach using infrared technique. Nevertheless, these methods still have some limitations, such as heavy computation and practical difficulties. In addition, we realized that digital navigation system provides better and clearer location information than paper map. Positioning system is not enough for practical use. Hence, these factors motivate me to design an indoor navigation system using image recognition. In this paper, we propose a method of an indoor navigation that is implemented on smartphone .The method uses the indoor environment features to locate user's location and a route calculating algorithm to generate an appropriate path for user. Users can know the direction indicated by a visual image immediately after the path is generated. The proposed method does not requires pre-installed tag or marker and hence its cost is much lower.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

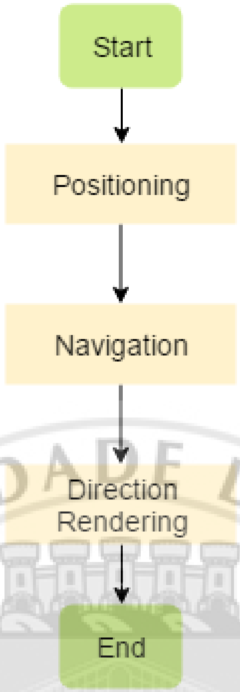
CHAPTER 1. INTRODUCTION

1.1 Overview

One of the most useful systems is navigation system which guides users to desired location. With the invention and popularization of mobile devices, navigation system has become handier and mobilized. The navigation system is mainly the outdoor navigation system nowadays. Global Positioning System (GPS) is the most appropriate navigation technology for outdoor environment. The GPS signals from satellite can be transmitted independently of any network signal. Therefore, smartphone user can be precisely located in the unobstructed outdoor environment. Nevertheless, it is not the case in the indoor environment. GPS relies on microwaves from satellite to locate users. Yet, microwaves are attenuated and scattered by obstacles in indoor environment and hence affects the performance of the GPS. Many researchers proposed various positioning methods such as Wi-Fi fingerprinting, dead reckoning, infrared technology and marker image recognition, etc. [1, 2]. However, part of the methods require pre-install equipment and maintenance. Moreover, network signals can also be easily affected in indoor environment with the presence of a great amount of obstacles in a tide area. Hence, these restrictions motivate me to define a new framework to overcome them for indoor navigation task.

This report is to propose a stable, easy-to-use offline indoor navigation method. The method can be divided into three phases: Positioning, Navigation and Direction Rendering. Figure 1 shows the overview of the method. The recognition process is done offline with image database stored on device in the positioning phase. Moreover, user's current location is identified in real-time using an efficient algorithm. Hence, pre-installed equipment is not required and the approach is convenient to be applied in different indoor locations.

Figure 1 Overview of proposed method



Route are calculated using the on-device map data in navigation phase. Hence no network connection is needed as well. Finally, an intuitive visual direction indicator shows the proper direction is integrated with the real world environment in direction rendering phase, leading users to their ideal destination.

1.2 Objectives

Due to the limitations of GPS in indoor environment, the aim of the proposed method is to define a new framework that is as reliable as GPS in outdoor environment. Moreover, the method should be applicable in any different indoor environments, such as shopping malls and museums, etc. These indoor environments share a common characteristic, there is no significant change in the environments within short period of time. Frequent update of the image database is not necessary. As the storage of smartphone is limited and not easy to increase, the expected size of the program is less than 200MB excluding the image database. The size of the database varies with different indoor environments, the size increases when the indoor environment is larger. However, only the feature of image is stored instead of the whole image in the database, the size should be small even though for larger amount of images.

The major difficulty to develop the proposed system is to find an appropriate image recognition algorithm. In indoor environments, such as shopping malls, the areas in such environment are always designed with the same design pattern, i.e. all the areas are very similar to each other. Hence, false positive rate will be very high if the recognition algorithm is not accurate enough. On the other hand, the algorithm is required to have high performance because the system is supposed to be used in real-time. A high accuracy and fast algorithm is very critical for the proposed method. Thus, the expected accuracy is 90% and the expected elapsed time for one image recognition process is less than 1 second. The reason for setting this expected elapsed time is based on the use situation. No rapid movement is expected when using the system and the image that needs to be recognized is not expected to change frequently. Hence, less than 1 second of elapsed time should be generally acceptable.

The proposed method not only overcomes the problems cause by the obstacles in indoor environment, but also makes use of the indoor environment characteristic to achieve the navigation task. In addition, the method does not required any hardware other than a smartphone, it significantly increases the feasibility of the generalization. Hence, the method has earned recognition and will be published in The 9th International Conference on Digital Image Processing (ICDIP 2017).

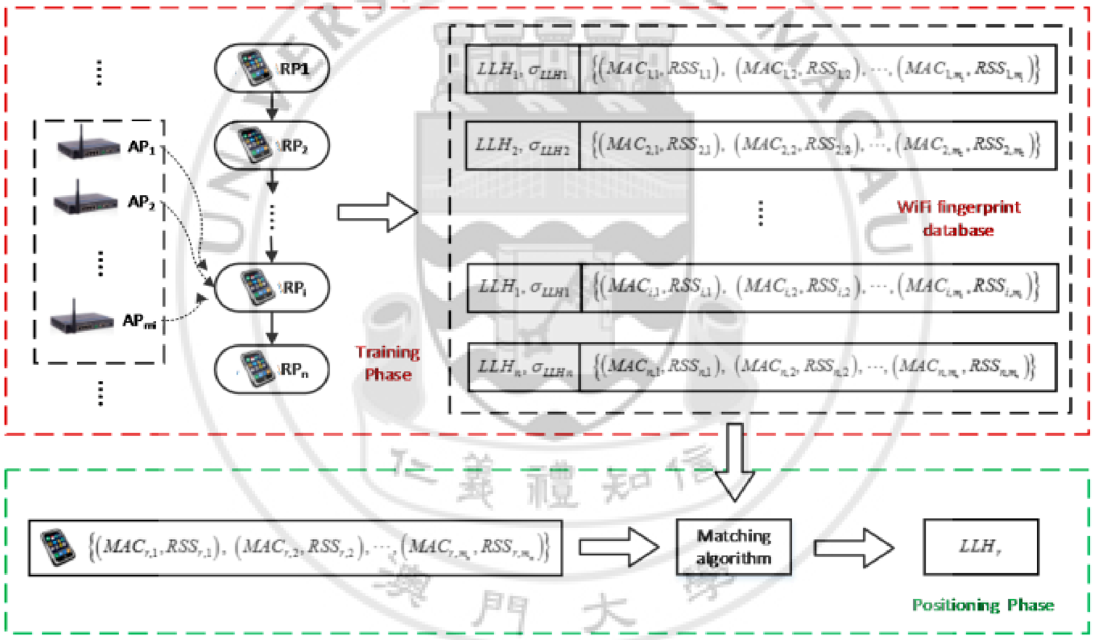
The following parts of this report is organized as follow. In chapter 2, different indoor positioning methods, their advantages and disadvantages are discussed. Functional requirement is specified in chapter 3. In chapter 4, the system design detail is specified. Implementation details including the used algorithm are in chapter 5. The test cases for the system, the corresponding results and discussion are in chapter 6. In chapter 7, the potential issues of ethic and professional conduct are discussed. Finally, I conclude in chapter 8. My contribution to this project is focused on the navigation phase. Hence, the design and implementation details are mainly discussed in this report.

CHAPTER 2. RELATED WORK

Navigation system can be divided into two parts: user's location detection and route representation. Many literatures have proposed different methods to detect the user's location. These methods can also be categorized into visual or non-visual method.

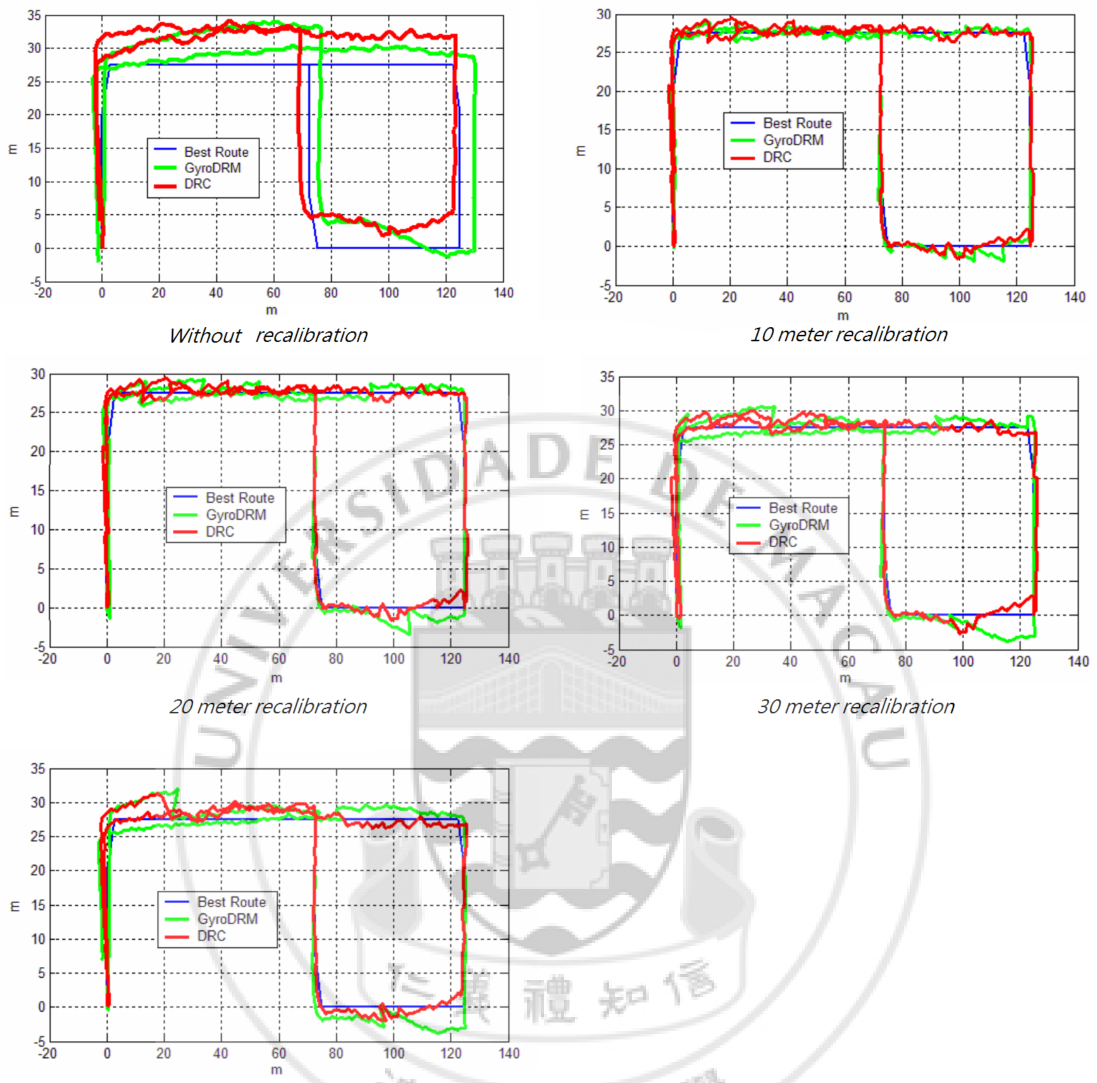
The non-visual methods are infrared technology, Radio technology and Wi-Fi fingerprint system [1], etc. Radio technology can provide non-visual tracking for the user by computing the distance between user and the signal station at known location. However, the signals maybe obstacle and not propagate in a way as expected. The systems Ultrasonic and Ultra-wideband can achieve high accuracy but the detections are expensive because they require more pre-installed equipment to increase the accuracy. Besides, a method that use Wi-Fi fingerprint to achieve positioning task is proposed in [3].

Figure 2 Procedure of Wi-Fi fingerprinting.



The procedure of positioning using Wi-Fi fingerprint is shown Figure 2, Firstly, the RSS from different Access Points are collected at different location using mobile device. Then, the collected RSS will be gathered to generate a database. This is the training phase and can be done without Wi-Fi connection. Then, the positioning phase can be done by comparing the received RSS and the RSS in database. Comparing with the radio technology, Wi-Fi fingerprint seems more stable. However, Wi-Fi connection is necessary for the navigation even though the database is trained offline. Moreover, more Access Points are needed in order to increase the accuracy.

Figure 3 Performance of two DR systems with different recalibration distance



Without recalibration

10 meter recalibration

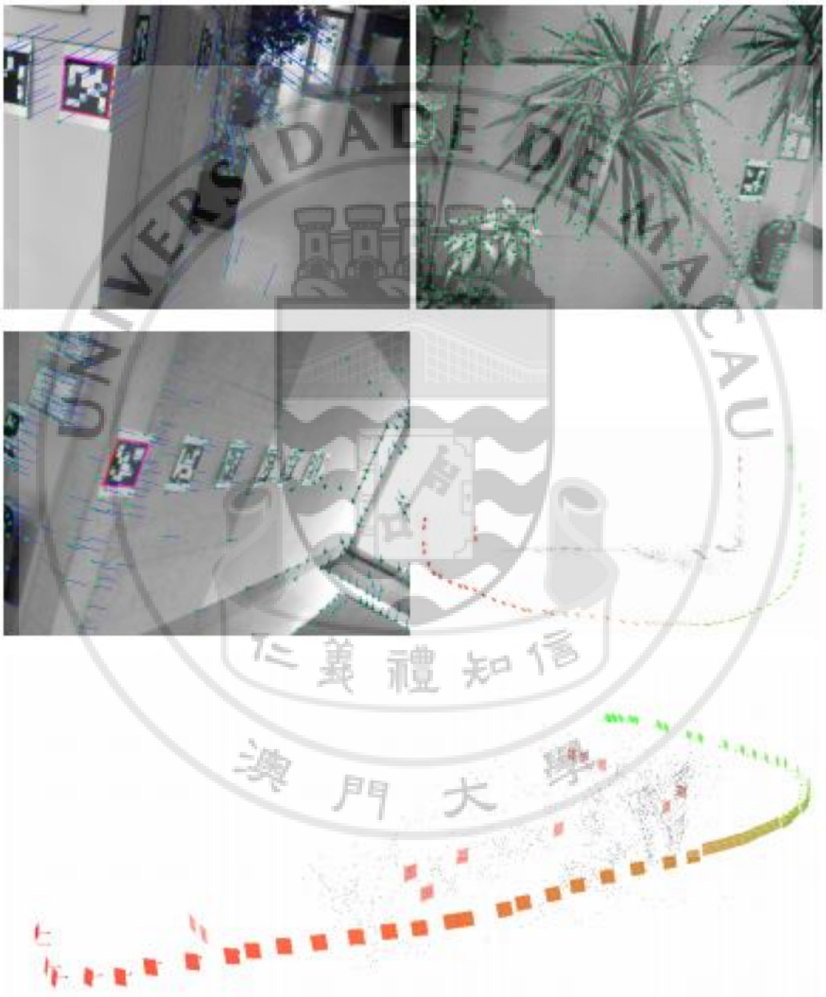
20 meter recalibration

30 meter recalibration

50 meter recalibration

D. Merico and R. Bisiani [4] proposed another non-visual method using Dead Reckoning (DR) technology to estimate the user's movement and direction. By computing the movement speed and direction, the distance between current location and starting point can be estimated. However, DR relies on the physical sensors, the accuracy can become very low when the sensors are not precise enough. Hence, DR requires periodically recalibrating [4]. Figure 3 shows the performance of 'Dead Reckoning Compass' (DRC) and GyroDRM which is gyro-aided dead reckoning module [4]. Without recalibration, the both of the systems have a significant difference with the best route. However, the performance of two systems are closer to the best route when the recalibration distance is less. Hence, DR can perform better when recalibration is more frequently. In contrast, visual methods are less costly. Camera is used to detect user's location by capturing photos. Therefore, visual method commonly uses marker-base system to achieve the localization. In marker-based system, Fiducial markers are usually used because it can achieve detection at a lower cost than non-visual method and easier to install. According to [5], continuous tracking of user's location is successfully achieve with large amount of Fiducial markers. The location information can be retrieved when the corresponding Fiducial marker is scanned with a camera. However, such implementation requires pre-marked Fiducial markers, i.e. the markers need to be printed and pasted on wall in the indoor environment. In Figure 4, many Fiducial markers are pasted in a small area in order to construct a 3D model. Therefore, calibration is harder to maintain when the area of indoor environment is larger. J. Kim and H. Jun [6] also proposed an alternative approach to track the user's position by taking photos of the current location. Then, the taken photos are compared to photos in online large-image database and retrieve location information. Obviously, network connection is required in such approach.

Figure 4 Images of location detection using Fiducial Markers



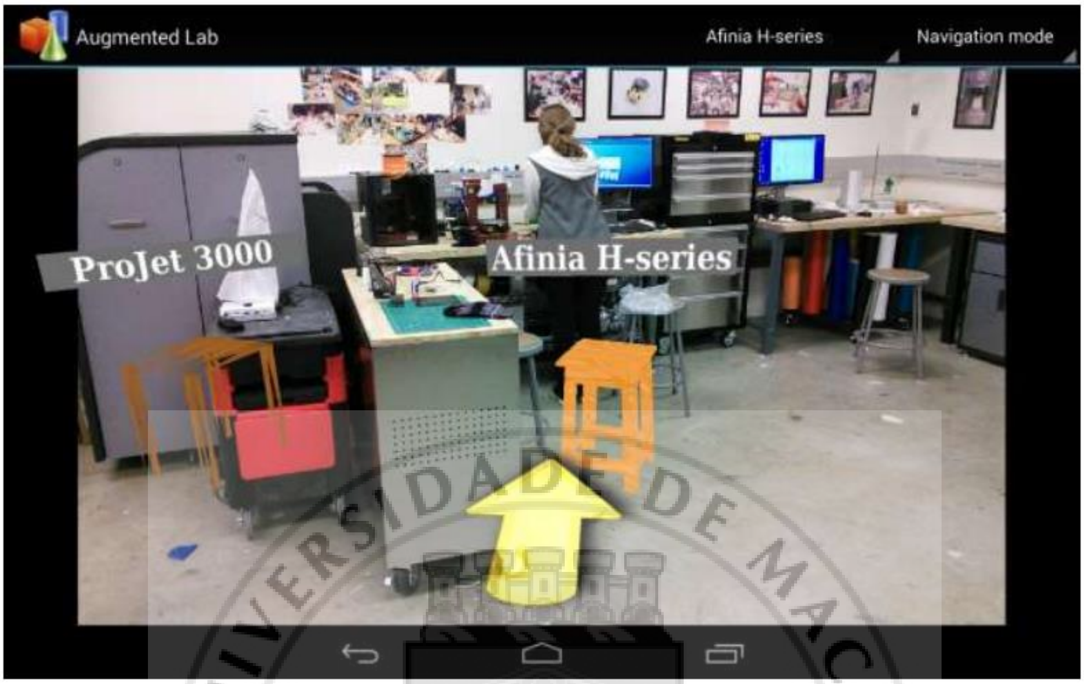
However, network signal is weak or even not available in some indoor environments. Even though in the case that network signal is available, mobile data is used to transmit images online database. The mobile data is usually limited monthly and exceeding the limit may be expensive for users.

Route representation, another crucial part in navigation system, mainly focuses on presenting the direction and route to the destination. Many literatures had proposed various representations for the navigation path. According to [7] , 2D or 3D map is not necessary for user using indoor navigation system since the sketch of direction is enough for the user. A. Mulloni, H. Seichter and D. Schmalstieg [8] using a sequence of activities to guide the user to destination, such as "walk 10 steps north", "walk 2 floors down" etc.

Figure 5 Screenshots of activity-based instructions system

This method could provide good and concrete instructions for short distance destination. Users can clearly know the distance to target location. However, if the distance between starting position and destination is longer, the number of activity will also be larger; user may be confused with those numerous instructions. In addition, A. Zhong [9] use an arrow to guide the user, which is a more straightforward method to show the direction.

Figure 6 Screenshot of using arrow in navigation system



Arrow has been used to indicate direction in daily life. For instance, showing the location of specific office or room, reminding driver to change the lane when a lane is being repairing, etc. Using arrow in navigation system can maintain the consistency of usage of arrow. Thus, users can easily be familiar with the system instantly.

CHAPTER 3. FUNCTIONAL SPECIFICATION

Even though the major purpose of this system is to guide user to the destination, the functional requirement of the system is not only limited to one. Other possible functions which are convenient for users should also be considered.

Table 1 Functional requirement description

Function	1. User shall be able to see a destination screen after the program has started up.
Input	---
Description	Once the user open the system, a destination screen which allows user to select the destination for the navigation shall appear. The destination screen shall be simple enough for user.
Output	A destination screen is displayed
Function	2. All the destinations shall be shown in the destination screen using a list.
Input	---
Description	In the system, all the information of shops in an indoor environment is stored on the device. Hence, in the destination screen, all the shops information are supposed to be displayed in the list view.
Output	A list view containing all destinations
Function	3. User shall be able to roll and check the destination list by slicing finger up and down in destination screen.
Input	Up and down movement on screen
Description	A list may not able to display all the shop information without rolling. Hence, the list shall be able to be rolled in order to show more shops information
Output	The list is rolled according to the user’s movement

Function	4. User shall be able to select the destination in destination screen.
Input	Click the item in the shop list
Description	In order to proceed to the positioning phase, a destination must be set. Otherwise, a route cannot be generated even though the user's location is identified. The shops in the list shall be able to be selected by clicking on the name or the radio button on the right of the shops name.
Output	The selected destination is highlighted.
Function	5. User shall be able to search for the destination by typing the name of destination in the textbox in destination screen. Unmatched destinations are filtered out.
Input	Text
Description	More the number of shop names, more difficult for user to select the desired shop. Hence, a textbox is placed on the top of the list. User shall be able to enter any text in the textbox but only the shop names which match the entered text is shown.
Output	A list of shop names which match the entered text
Function	6. User shall be able to set the destination and a navigation screen shall appear by clicking on 'Go!' button in the destination screen after one destination is selected.
Input	'Go!' button is clicked
Description	A 'Go!' button which is disabled is placed below the shop list. Once the user selects one destination, the button become enable for user to click. The selected item is set to the destination and positioning phase starts.
Output	A navigation screen appears

Function	7. User shall be able to switch to the destination screen and reselect the destination by clicking on the ‘Destination’ button in navigation screen.
Input	‘Destination’ button is clicked
Description	In the navigation screen, user shall be able to click the ‘Destination’ button which is located at the corner of the screen. Once the button is clicked, the destination screen shall appear for user to select a new destination.
Output	Destination screen is displayed
Function	8. A direction indicator shall appear to indicate the direction that user should follow after the current location is identified in the navigation screen.
Input	Captured real time image with camera
Description	Once the system identifies the user’s location using image recognition, a shortest route to destination is generated. A direction indicator shall appear and indicate the direction that user needs to follow in order to reach the destination.
Output	A direction indicator which shows the direction
Function	9. An arrival message shall appear in the navigation screen when the user has reached the selected destination.
Input	Captured real time image with camera
Description	For each captured image, the system checks whether it is the image of destination. An arrival message shall appear to inform user that the destination has reached when the checking result is positive.
Output	An arrival message

CHAPTER 4. SOFTWARE DESIGN SPECIFICATION

My proposed method can be divided into three phases: positioning, navigation and direction rendering. According to the objectives we set up, the system shall include image recognition process which is done offline. Moreover, the process shall be done efficiently according to my expecting accuracy and performance of this system. Hence, the major design effort of this system is to focus on the positioning phase. In the positioning phase, suitable algorithms for feature extraction and feature matching are needed. Instead of inventing new algorithms, examining and choosing existing algorithms is more appropriate for this project since my goal of this project is to define a new framework rather than researching on algorithm. The user's current location information is obtained by image recognition process in this phase.

In the navigation phase, a best and shortest route is generate from user's current location and destination. Hence, the effort in this phase is to design a user friendly interface for user to input the destination information. In addition, an algorithm to generate the route needs to be chosen. The data structure to represent the geographical situation of the indoor environment also need to be designed.

Finally, the direction rendering phase focus on direction calculation and direction representation. From the generated route, the system shall be able to calculate the direction that user should follow at user's viewpoint. Thereafter, an intuitive direction indicator shall be designed. Thus, the system can provide a consistence feel to user and does not require user to learn how to use the system. User only need to follow the direction indicated by the indicator.

4.1 System Architecture

Figure 7 Work flow of the proposed system

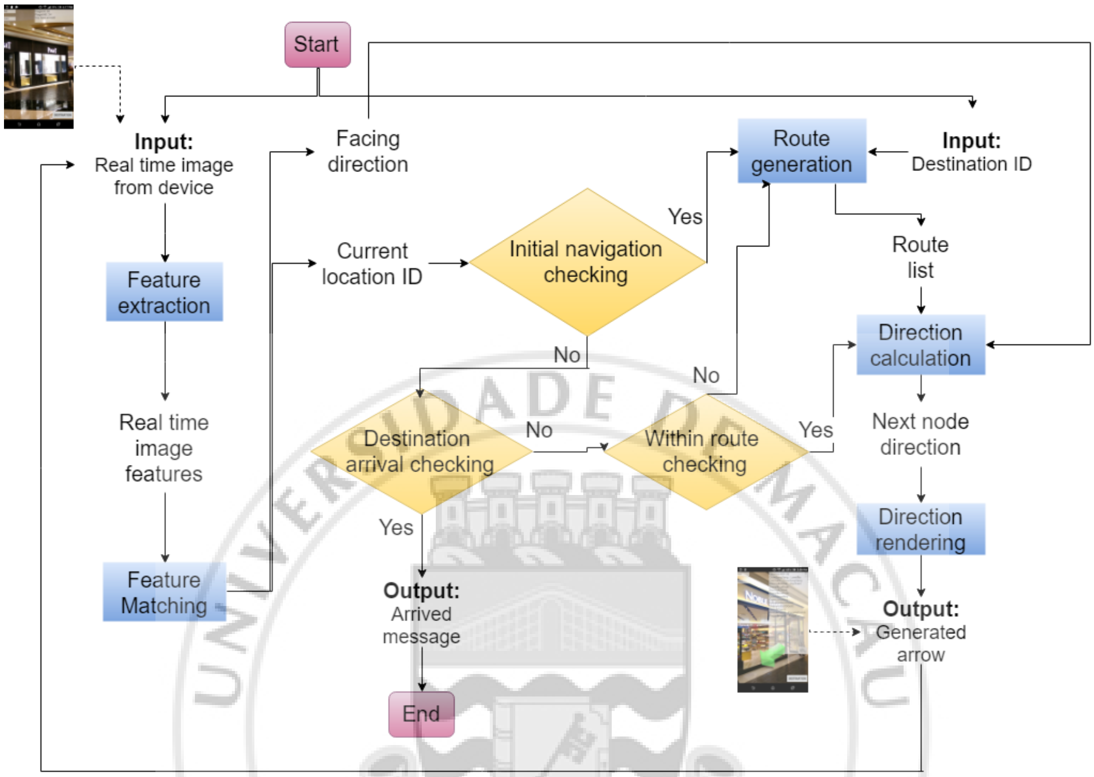


Figure 7 shows the workflow of the system. The system can be divided into 3 phases: positioning, navigation and direction rendering. Firstly, user needs to select the destination in a destination screen. Secondly, the positioning phase is started. The system keeps capturing images of user's location. Image recognition is done on each captured image. The location information is identified and passed to the navigation phase. From the destination and current location, a shortest path is generated with geographical information of the indoor environment. The path is then passed to the direction rendering phase. In the direction rendering phase, the indicating direction at user's viewpoint is calculated with geographical information. Finally, the calculated direction is shown with intuitive representation. The system keeps repeating these three phases until the user reaches the destination at the navigation phase.

Figure 8 Dataflow diagram of the system

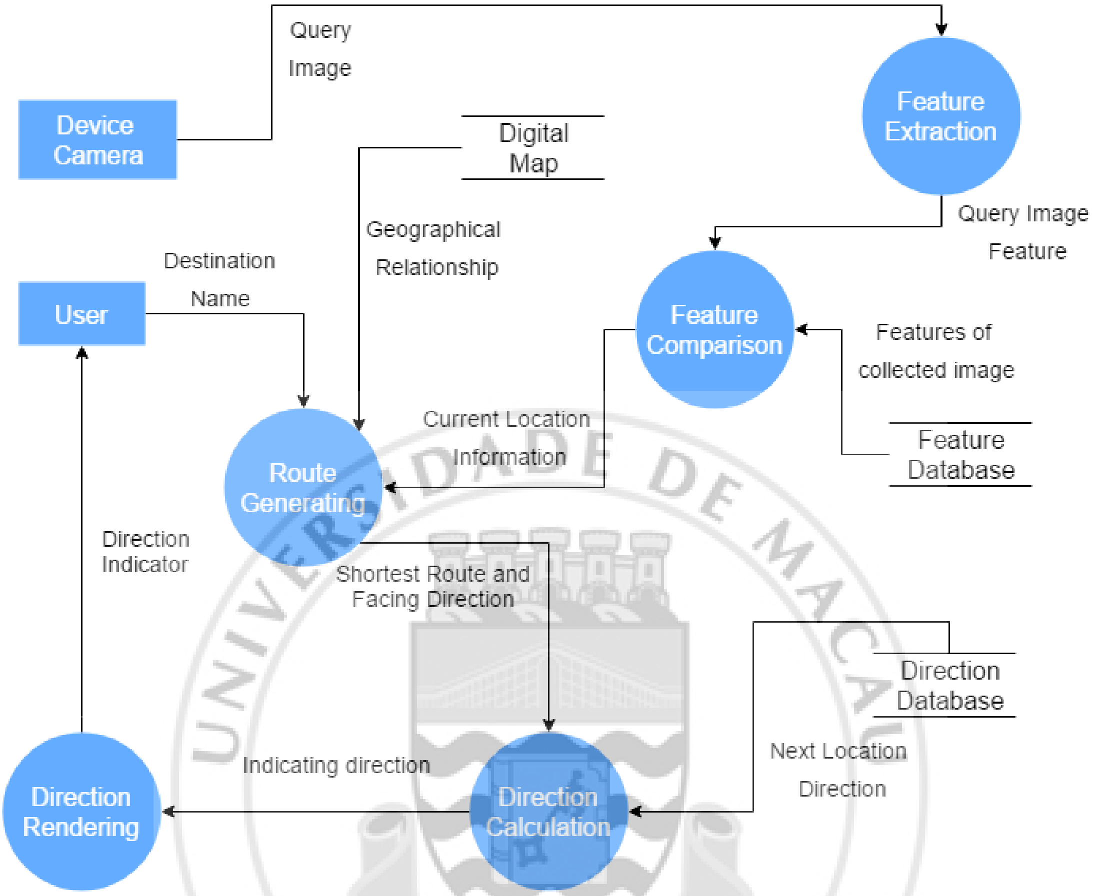


Figure 8 shows the data flow in the system and provides clearer concepts of some processes. Firstly, the system pass the query image captured using device camera to the feature extraction process. Secondly, the query feature is then passed to the feature comparison process. The features of collected image are stored on the device and fetched in this process. By the comparison, the current location information is obtained using the collected feature that is the most similar feature to the query feature. Thirdly, a shortest route can be generated using the geographical relationship of digital map. The map is made and installed in advance instead of at the run time. Thereafter, the next location information can be know from the shortest route. Hence, the direction of next location at user's viewpoint is calculated using the facing direction and absolute direction of next location obtained from direction database. Finally, the direction indicator is presented for user.

Figure 9 State diagram of the system

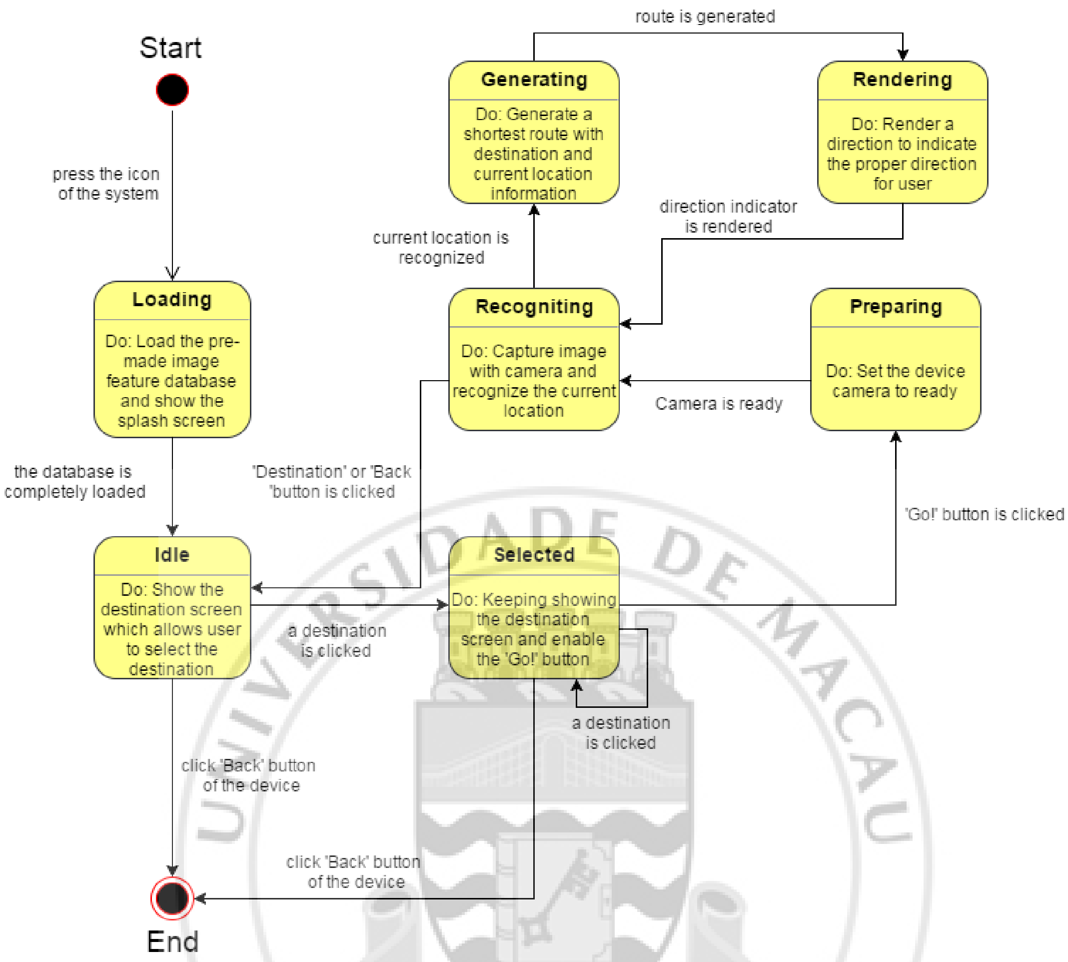


Figure 9 shows the state diagram of the system. The interaction between user and the system can be easily understood with this diagram. Once the user starts the system, the system first loads the feature database. The system is in 'Idle' state after the database is completely loaded. The system changes to the 'Selected' state when user selects a destination. In both of the 'Idle' and 'Selected' state, user can end the system by clicking the 'Back' button which is a functional button in Android system. The system changes to 'Preparing' state and set the device camera to ready when the 'Go!' button is clicked in 'Selected' state. After the camera is ready, the systems loops the 'Recognizing', 'Generating' and 'Rendering' states until user clicks 'Back' and 'Destination' button. Those three states are corresponding to the three phases of the proposed method and the 'Destination' button is at the corner of the navigation screen.

4.2 Navigation

In navigation phase, the only thing needs to do is to generate a shortest path between user's location and destination. However, the real-time images are continuously captured. Resources are wasted when route is generated under unnecessary conditions.

Figure 10 Workflow of navigation phase

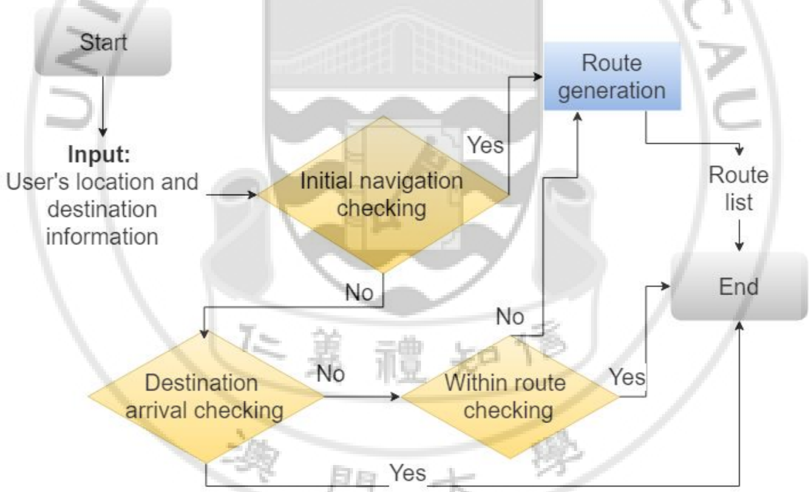


Figure 10 shows the workflow of navigation phase. With the inputs obtained from positioning phase, the system checks whether this execution is the first execution of the system. There is no any generated route if it is the first execution of system. Hence, it is necessary to generate the shortest route at this situation. Thereafter, the route is passed to next phase to render a direction indicator. Once the direction rendering phase has completely, one cycle of the system has finished. The system keeps cycling until the destination is reached. In the second or later execution, the system still checks whether it is the first execution. Obviously, the result is negative and the destination arrival checking is done. The system directly shows an arrival message and stop cycling if the result of the arrival checking is positive. Otherwise, the final checking is done in order to determine whether user follows the generated route. For some reasons, user may not always follow the provided route. In such cases, a new route needs to be generated according to the new location of user. However, to generate a new route in every execution wastes the computational resources. Hence, a within route checking is done in the system. The system ends the navigation phase directly and proceeds to direction rendering phase to refresh the direction indicator if the result if positive. Otherwise, a new route is generated.

In short, computational resources can be saved under some conditions even though only one thing needs to be done in navigation phase. When the system is in first execution or user does not follow the route, a new shortest route is generated. In other cases, the destination is reached or user follows the route, the route generation process is skipped and the system ends or proceeds to the direction rendering phase.

CHAPTER 5. IMPLEMENTATION NARRATIVE

5.1 Navigation

5.1.1 Data structure and algorithm

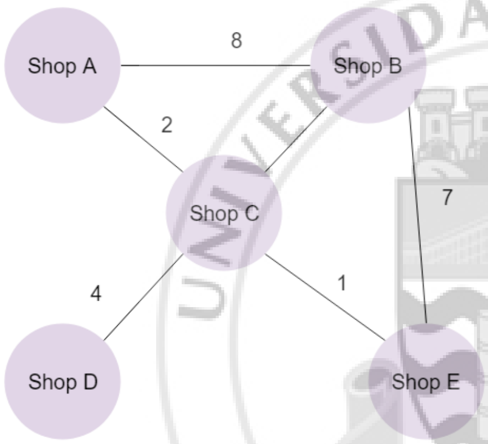
After confirming the position of user by image recognition, it is necessary to calculate the route from the position to the pre-input destination. The map data needs to be stored on the device thus the route can be calculated offline. Below is a table showing five candidates of abstract data type used to implement the map data of indoor environment.

Table 2 Description of different Abstract Data Types

ADT	Description
Singly Linked List	A sequence of nodes. All nodes are connected and the sequence can only be visited in single direction, i.e. a visited node cannot be revisited again unless the whole sequence is visited. Node can be dynamically instead of statically inserted or deleted during the execution and hence saving the memory.
Doubly Linked List	Similar to the singly linked list but the sequence can be visited in two direction, i.e. a visited can be revisited again even though the whole sequence has not been visited yet.
Stack	The nodes are in Last-In-First-Out order. Pop and push are two typical operations. Pop can delete the latest added node and push can add a new node at the top of the stack. Stack is easy to be implemented and interpreted.
Graph	Nodes in graph is interpreted as vertices. Edge is to connection two vertices. Graph can be directed or undirected, weighted or unweighted. Graph is also simple to be interpreted but it is time consuming and costly to produce a graph.
Tree	One node at the top is called root. Each node has two or more nodes which are called children. Tree can have a good performance when each node has a measurable value. Search, insertion or deletion for specific value is efficient.

From the Table 2, the characteristics of different ADT are summarized. Graph is selected for the map data implementation. More specifically, weighted undirected graph is the best candidate for storing map data. Weighted undirected graph provides a flexible way to visit different nodes. There is no visiting order in graph as long as vertices are connected with edges. Operations in other ADTs need to follow the visiting order which matches the characteristic of ADT. Nevertheless, it is time consuming and costly to build a graph at run time. Hence, a pre-made map data in order to minimize the negative influence of using graph. The indoor environment can be divided into many small areas. One single vertex represents one of the areas. The weight of edge between two vertices is assigned according to the geographical distance between the corresponding two areas. Then, the digital map of the indoor environment can be implemented when all the areas are correctly connected and the weight between two vertices are precisely assigned.

Figure 11 Map data implementation using Weighted Undirected Graph



Shop name	weight	Con_shop_name
A	8	B
A	2	C
B	7	E
C	1	E
D	4	C

Figure 11 shows the implementation of map data stored database and the visualized graph. In addition, an existing single-source shortest-path algorithm - Dijkstra's algorithm can be applied to find the shortest path. The following are the steps of Dijkstra’s algorithm.

1. Set initial node to 0 and to infinity for all other nodes.
2. Set the initial node as current. Mark all other nodes unvisited
3. Consider all unvisited neighbors of current node and calculate their tentative distances
4. Mark the current node as visited
5. If the destination node has been marked visited, the algorithm has finished
6. Otherwise, select the unvisited node with the smallest tentative distance, set it as the new "current node", and go back to step 3

The idea of this algorithm to find a shortest is to travel all the possible paths from starting point to destination point efficiently. Once all the paths are travelled, the total weight of travelling the corresponding path is calculated. The shortest path is the path with lowest weight. For any graph, the upper bound of the running time is $O(|V^2|)$ where $|V|$ is the number of vertices.

5.1.2 Coding

In the navigation phase, the major thing needs to do is to calculate the shortest route from user's location to destination. A tool called JGraphT which is implemented using Java is used in this phase. An undirected weighted graph, which is treated as digital map in the system, can be made with a list of vertices and edges using JGraphT. Figure 11 also shows the example of the list. Each entry represents one edge and a weight is assigned according to the geographical distance. In order to minimize the computational cost at run time, the map data is built and installed in advance, rather than at the time of execution of the program. Hence, using the Dijkstra's algorithm implemented in JGraphT, a shortest route is calculated using the location id and destination id. Then, the route is stored as a list and passed to the direction rendering phase to calculate the direction needs to be rendered. The following is the pseudo code of route calculation.

```
private String CalculateRoute() {

    route = new DijkstraAlgorithm(MapGraph, CurrentID, DestinationID);
    String resultsText = "The cost is: " + route.Length() + "\n";
    for every edge in the shortest route {
        print(the current area and the next area)
    }
    return resultsText;
}
```

Basically, the route is generated using the interface provided by the JGraphT. Moreover, there are some attributes such as the length and the complete route can be used for debugging.

As mentioned in 4.2, a route does not need to be generated under some conditions. The following is the pseudo code to determine whether the system is in such conditions.

```
//first navigation request
if (NextDirection.isEmpty)
    CalculateRoute();
```

The system checks whether it is the first execution by checking the variable 'NextDirection' is empty. This variable is assigned a value in the direction rendering phase. Hence, nothing is assigned in the navigation phase of at the first execution. Otherwise, the system checks whether the user has reached the destination.

```
//arrived
else if (CurrentPlaceID.equalsTo(DestinationID)) {
    Toast.makeText("You have arrived!").show();
    StopRecognizing();
}
```

It is done by comparing the current location ID with the destination ID. An arrival message is shown on the screen of the device if they are identical. Then, the within route checking is done by comparing the current location ID with 'PrevPlaceID' and 'NextPlaceID'.

```
//next area has not been reached yet
else if (CurrentPlaceID.equalsTo(PrevPlaceID)) {
    ReFreshDirection();
```

```
}

//next area has been reached
else{
    //not in route
    if (!CurrentPlaceID.equalsTo(NextPlaceID)){
        CalculateRoute();
    //in route
    }else{
        getNextDirection();
        ReFreshDirection();
        PrevPlaceID = CurrentPlaceID;
    }
}
```

The ‘PrevPlaceID’ is assigned the current location ID at the end of direction rendering phase. Then, ‘PrevPlaceID’ is compared with the new obtained current location ID in the next cycle of system. The same of both variables represents user remains in the same area or has not left the area yet. This condition is considered as following the route and no new route is generated. If the ‘PrevPlaceID’ is not the same as current location ID, it represents the user has left the previous area. The ‘NextPlaceID’ is assigned the ID of next area when a new route is generated. By comparing the ‘NextPlaceID’ and current location ID, the same of both variables represents user follow the indicated direction and no new route is generated. Otherwise, it represents user has left the area to a new area which is not in the route. Therefore, a new route needs to be generated because of the arrival of unexpected area.

CHAPTER 6. SYSTEM QUALITY

6.1 Test Environment

In the section, two test cases are used to test the implemented. The following table is the detail of the testing environment.

Table 3 Detail of testing environment

Location	A shopping mall in Taipa
Area(in m^2)	~30000
Number of divided area	79
Number of collected images	~700
Device	HTC M8
System version	Android 6.0

One shopping mall in Macau Taipa is selected to be the testing indoor environment. The total area of the shopping mall is around 90000 m^2. One third (~30000 m^2) of the total area is chosen for testing. The chosen area is divided into 79 small areas. For each small area, at least 8 images are taken to build the collected image database. Basically, for each area, 1 image is taken for 1 direction and hence 8 images are taken for 8 directions. However, additional images are taken for some areas which are the intersection points of different areas or geographically larger than other areas. Therefore, the collected image database contains around 700 image features for those 79 small areas. The system is implemented on Android 6.0 and tested with HTC M8.

6.2 Positioning Test

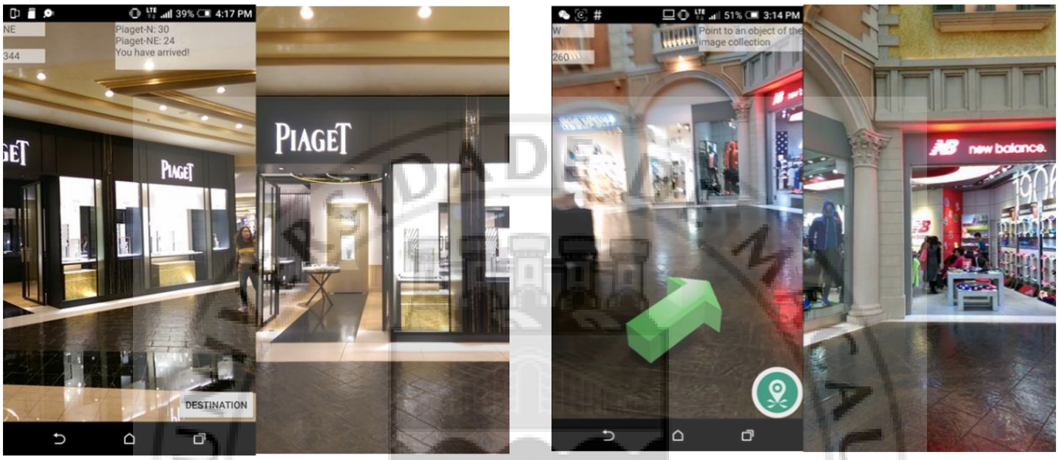
In the positioning test, 20 testing areas are randomly picked. For each testing area, 8 trials are done. Tester stands at a random position within that area and checks whether the system can recognize the current location within 1 second.

Table 4 Recognition testing result

Location	**1st**	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**	**Success Rate(%)**
Shop A	✓	✗	✓	✗	✓	✓	✓	✓	75
Shop B	✗	✓	✓	✓	✓	✓	✓	✓	87.5
Shop C	✓	✓	✓	✗	✗	✓	✗	✓	62.5
Shop D	✓	✓	✗	✓	✓	✓	✓	✓	87.5
Shop E	✓	✓	✓	✓	✗	✓	✓	✓	87.5
Shop F	✓	✓	✗	✓	✓	✗	✓	✓	75
Shop G	✓	✗	✓	✓	✓	✗	✓	✓	75
Shop H	✓	✓	✓	✓	✓	✗	✗	✓	75
Shop I	✓	✓	✗	✓	✓	✗	✗	✓	62.5
Shop J	✗	✓	✓	✓	✓	✓	✗	✓	75
Shop K	✓	✗	✓	✓	✓	✓	✓	✓	87.5
Shop L	✓	✓	✓	✓	✓	✓	✓	✗	87.5
Shop M	✓	✓	✓	✓	✗	✓	✗	✓	75
Shop N	✓	✓	✓	✓	✓	✗	✓	✓	87.5
Shop O	✓	✓	✗	✓	✗	✓	✓	✓	75
Shop P	✓	✓	✗	✓	✗	✓	✓	✓	75
Shop Q	✓	✓	✗	✓	✗	✓	✓	✓	75
Shop R	✓	✓	✓	✓	✓	✓	✓	✓	100
Shop S	✓	✓	✗	✓	✗	✓	✓	✓	75
Shop T	✓	✓	✓	✓	✓	✗	✓	✓	87.5
Average Success Rate(%)									79.37

Table 4 shows the testing result of the position recognition in different location. '✓' indicates the trial is successful and the '✗' indicates the trial fails. From the table, it shows that the success rate of most of the testing areas are 75%. Yet, two of the areas have a success rate of 62.5%. For such a low success rate, there are two reasons of the recognition failures. The first one is the query image is at the viewpoint that is are two reasons of the recognition failures. The first one is the query image is at the viewpoint that is extremely different with the reference images. When the tester captures the query image at a very different viewpoint, such as the border of the area, from reference image, the query image contains fewer common features of the reference image. Figure 12 shows the success and failure pairs of recognition. In the success pair, left image is the query image and the right is the reference image. There are many common elements in two images. For instance, logo of the brand, windows and the whole front view of the shop, etc.

Figure 12 Comparison of success and failure recognition



Success pair

Failure pair

Nevertheless, the common elements in the failure pair are much fewer. In the reference image, the view inside the shop and the logo of the shop is clearly captured. Yet, none of the view inside the shop or the logo is clearly captured in the query image. Hence, too few of common elements lead to the recognition failure.

The second reason is related to the crowd. A great amount of people enter the view of the camera and block most of the features of the query image. The captured image contains many elements of the crowd rather than the elements of shop. Thus, the query image is very dissimilar with the reference image and fewer common features. As a result, the recognition performance is lowered and the average success rate to 79.37%. Nevertheless, this testing focuses on examining the recognition performance and it cannot represent the navigation performance. The navigation performance is shown in the following section and some improvements of the recognition will be discussed in last section of this chapter.

6.3 Overall System Test

In the overall system test, 10 testing starting location and destination pairs are randomly picked. For each pair, 10 trials are done. Tester needs to follow the direction indicated by the system and check whether the destination can be reached under the guidance of the system. A sudden change of the indicated direction is considered a failure. Sudden change means the indicated direction changes back and forth.

Table 5 Navigation testing

From	A	Q	R	L	D	M	S	C	F	B	H
To	T	B	H	E	Q	A	E	S	O	I	T
Average success rate	80 %	90 %	100 %	90 %	100 %	100 %	100 %	80 %	100 %	100 %	100 %
Total Average Success Rate									95%		

Table 5 shows the overall system test result of the proposed system. The testing result shows that the total average success rate is 95%. The reasons of some pairs have success rate of 80% or 90% are the same as the positioning test. The first reason is the dissimilarity of reference and query image. The second reason is the crowd blocks the camera view. Both of the problems affect the overall system test as well. In addition, one more reason for the failure of the overall system test. The test fails due to the user's movement. During the testing, tester holds the device with hands and walks to the destination. As a result, the tester's movements also cause the device to vibrate and lower the quality of captured real-time images. The details of the query image are blurred, resulting an increase the dissimilarity between the query image and the reference image. Hence, the system incorrectly or does not even identify the current location and causes the testing to fail.

On the other hand, the reason of such high success rate is that in the case if recognition failure, the device sensors were used as a backup facing direction calculator. As mentioned before, the user's facing direction is originally obtained by image recognition. However, the image recognition may fail due to the reasons mentioned above. The facing direction is calculated from the device sensors value when the recognition failure occurs. Hence, system still able to indicate a proper direction to user by using the calculated facing direction to render a new direction indicator. Nevertheless, the calculated facing direction can only be used within one area. For instance, the failure of image recognition happens in area A, the calculated facing direction can only be used until the next area, for example B is reached. In the area B, user may need to turn left but the system cannot identify the area B and hence not able to render a direction indicator that point to left. Therefore, the system needs to successfully identify the new area when it is reached. Yet, this case rarely happens due to the good performance of the image recognition process.

6.4 Discussion

For the proposed method, the system quality is evaluated by validation. In the positioning test, the average success rate is around 80%. Two reasons are due to the image recognition failure, one is the viewpoint difference of the query image and reference image, another one is the features of the image are blocked by the crowd.

Basically, the common thing of the reasons is the dissimilarity of reference image and query image. Hence, the success rate can be increased by collecting more reference images. For example, failure pair shown in Figure 12, the recognition will succeed if the reference image is the same as or similar to the query image. By collecting more reference images, different viewpoint images are captured and hence less dissimilarity between the reference images and possible query images. Even though the dissimilarity is due to the crowd, the success rate can also be increased by capturing image with crowd. By comparing the reference image which is full of people, the query image can possibly be properly recognized. Even though the recognition is no guaranteed, the possibility of successful recognition is increased with the similar reference image. Thus, the number of reference image determines the success rate of the positioning phase.

Table 6 Existing indoor navigation methods and proposed method comparison

	Image recognition	Wi-Fi Fingerprinting [12]	Image Marker [13]
Continuous Localization	✓	✓	✗
Network Requirement	✗	✓	✗
Positioning Constraint	✗	✓	✓
Require Pre-installed equipment	✗	✓	✓
Error Rate	5%	10%	2%

In the overall system test, the system is able to recognize the current location using device camera, calculate the shortest route from current location to destination and provide proper direction while user is walking to the destination. The expectation of this project is to develop a high accuracy and high performance offline navigation system. From the testing results, the accuracy of the navigation system is 95%. In addition, there is no restriction on the user during the use of the system, such as keeping the device stationary or facing the device to the logo of a shop, to carry out the recognition process. In addition, the system is able to identify the current location within 1 second in most of the cases. Hence, the system is considered to be successful and meets the expectation.

Table 6 shows the comparison of the proposed method with other indoor navigation method. Both of image recognition and Wi-Fi fingerprint are able to locate user in everywhere of an indoor environment. Yet, image marker when some specific points are reached. Image recognition does not require network connection and pre-installed equipment. Moreover, there is no restriction on user during the user of system using image recognition. Image marker method requires pre-installed markers and user needs to discover and scan the markers. Wi-Fi requires electromagnetic signal that are easily affected in indoor environment. Hence, pre-installed equipment is needed to emit signal and the location can only be identified within the effective range of the equipment. The proposed method does not have such constraints and result in a low error rate. This comparison shows the proposed method is better than other existing methods. The proposed method overcomes the disadvantages of other methods. Moreover, it gives an intuitive direction indicator which is user friendly. The only thing that user needs to do is to follow the guidance of the system. However, one improvement could be made is the representation of the direction. Currently, the system shows the direction using an arrow image with an 8-azimuth angle. A continuous representation that similar to a compass can provide a better user experience and clearer direction information.

CHAPTER 7. ETHICS AND PROFESSIONAL CONDUCT

7.1 Portrait Right

When implementing the system, numerous photos of the indoor environment are taken. The photos not only include the logo of shops, decorations of the indoor environment but also the faces of people such as shop keepers, security guards and customers. In the Honour Project, all the collected photos will not cause any legal problems since the Honour Project is academic. The portrait of someone can be legally used without permission of portrait owner for academic purpose. However, the implemented system has a potential to be extended to commercial use. Similar to this case, Google Map contains huge amount of photos which contains people's portrait. Yet, the purpose of the photos is to view rather to calculate. Hence, part of the photos which contains someone's portrait can be blurred. On the other hand, the photos in the proposed method need to be recognized. Any changes on the collected image may affect the feature extraction process and hence lower the overall performance. Fortunately, the photos only need to be kept temporarily. They can be deleted after the feature extraction process. Any update of the collected image database can be done by capturing new photos of specific area and applying feature extraction algorithm again. Therefore, all the collected image in this project are deleted.

7.2 System use and sneaking shot

When using the system, user needs to hold the device horizontally. The pose of using the system is exactly the same as taking photos. Hence, a sneaking shot problem appears when the system is used in daily life. Sneaking shot is an action that describe someone secretly taking photos of other people. When we were doing the overall system test, people thought that we were taking videos or photos and walked away in order not to be captured. In contrast, if the system is released to the public, someone will be considered as using the system while the device is being held horizontally. However, that person may actually take sneaking shots. Thereafter, the indoor environment in which the system can be used may become the hotbed of sneaking shots. Moreover, sneaking shots problem may also relate to the other problems, such as the portrait right problem mentioned above. Thus, it is a complicated problem needs to be address when the system in widely used. Fortunately, the sneaking shots problem has not yet arose since we are the only people using this system. Yet, the differences of sneaking shot and normal use of the system needs to be distinguished when the system in released to the public.

CHAPTER 8. CONCLUSIONS

In this paper, an image-based indoor navigation application is proposed to overcome the navigation challenges in indoor environment. GPS signals are scattered and attenuated and hence GPS is not suitable in indoor environment. Other indoor navigation methods are proposed by some researchers. However, those methods have some limitations lead to the less viability. Therefore, we proposed a method which achieve the indoor navigation task using image recognition. The method is classified into three main phases: positioning, navigation and direction rendering. Positioning is achieved by image recognition using DARTs descriptor. DARTs descriptor has a low computation cost and suitable for real-time processing. REVV is used to compare the difference between DARTs descriptor of query image and all the descriptor in the image database. Dijkstra‘s algorithm is used to perform the shortest path search on the indoor map graph and return a route to destination. Repeated positioning processes and route updating are performed during the navigation. Direction rendering integrate an arrow image to indicate the direction and the real environment on screen to provide user-friendly experience and increase acceptability of the application.

The accuracy of the proposed navigation method is demonstrated in the system Quality chapter. In terms of continuous localization and unconstrained condition, the proposed method have better performance compare with other indoor navigation methods. The advantage of the proposed indoor navigation method is totally free of internet connection and the accuracy will also not be affected by attenuation of electromagnetic waves caused by obstacles. The accuracy can be improved by collecting more reference images. Hence, the future improvement could focus on collecting more image data to increase the success rate, reduce recognition time and include additional information of the indoor environment in the process of navigation. One of the common problems of smartphone application is the battery usage. The system requires continuous computations in order to recognize the current location. Thus, a balance between the performance and the battery usage in order to increase the endurance. In addition, the problem of crowd also needs to be considered. Crowd is inevitable in indoor environments and the system should also work properly in such situation.

REFERENCE

[1] R. Harle, "A Survey of Indoor Inertial Positioning," in *IEEE Communications Surveys & Tutorials*, 2013.

[2] H. Liu, H. Darabi, P. Banerjee and J. Liu, "Survey of Wireless Indoor Positioning," in *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2007. pp. 1067-1080*, 2007.

[3] P. Zhang, Q. Zhao, Y. Li, X. Niu, Y. Zhuang and J. Liu, "Collaborative WiFi Fingerprinting Using Sensor-Based Navigation on Smartphones," in *Sensors, vol. 15, no. 7, pp. 17534–17557*, 2015.

[4] M. Davide and B. Roberto, "Indoor Navigation with Minimal Infrastructure," in *Positioning, Navigation and Communication: 4th Workshop*, 2007.

[5] K. Manfred and S. Dieter, "Automatic Reconstruction of Wide-Area Fiducial Marker Models," in *Mixed and Augmented Reality 6th IEEE and ACM International Symposium*, 2007.

[6] J. Kim and H. Jun, "Vision-based location positioning using augmented reality for indoor navigation," in *IEEE Transactions on Consumer Electronics*, 2008.

[7] C. Kray, K. Laakso, C. Elting and V. Coors, "Presenting Route Instructions on Mobile Devices," in *Proceedings of the 8th international conference on Intelligent user interfaces*, 2003.

[8] A. Mulloni, H. Seichter and D. Schmalstieg, "Handheld Augmented Reality Indoor Navigations with Activity-Based Instructions," in *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, 2011.

[9] A. Zhong, "Improving User Experiences in Indoor Navigation with Augmented Reality," 2014.

[10] D. Marimon, A. Bonnin, T. Adamek and R. Gimeno, "DARTs: Efficient scale-space extraction of DAISY keypoints," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference*, Barcelona, Spain, 2010.

[11] D. M. Chen and B. Girod, "Memory-Efficient Image Databases for Mobile Visual Search," in *IEEE MultiMedia*, 2014.

[12] F. Adel, M. Makary and M. El-Nahas, "Indoor Positioning and Navigation System for Interior Design Augmented Reality," in *The Tenth International Conference on Digital Telecommunications*, Barcelona, Spain, 2015.

[13] B. A. Delail, L. Weruaga and M. J. Zemerly, "CAViAR: Context Aware Visual Indoor Augmented Reality for a University Campus," in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012 IEEE/WIC/ACM International Conferences*, 2012.

[14] C. Randell and H. Muller, "Low Cost Indoor Positioning System," in *Ubicomp 2001: Ubiquitous Computing. UbiComp 2001*, 2001.