Smart Navigation System - Building Recognition Server Development

BY

CHIN CHEE YANG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology

(Perak Campus)

JANUARY 2015

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Smart Navigation System - Building Recognition Server Development

Academic Session: January 2015

I ____________________ CHIN CHEE YANG ____________________
(CAPITAL LETTER)

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

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

Verified by,

_________________________ _________________________

(Author's signature) (Supervisor's signature)

Address:
85, Jalan Pusing
Papan Baru
31550 Pusing, Perak.

____
Supervisor's name

Date: ____________________ **Date**: ____________________

Smart Navigation System - Building Recognition Server Development

BY

CHIN CHEE YANG

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONS)

Faculty of Information and Communication Technology

(Perak Campus)

JANUARY 2015

DECLARATION OF ORIGINALITY

I declare that this report entitled "**Smart Navigation System - Building Recognition Server Development**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ________________________

Name : ________________________

Date : ________________________

ACKNOWLEDGEMENTS

Without the contribution of the following important person, this final year project would not successfully completed on time. Hence, I would like to take this opportunity to extend my gratitude to them for their contribution in completing this project.

First of all, I have to thank my final year project supervisor, Dr Ng Hui Fuang for supervising me throughout this project. Without his guidance and dedicated involvement in every details throughout the process, this report would not been completed. His knowledge and experiences enables me to have a deep understanding on computer vision which helped me a lot in developing this system. I too would like to thank him for his support and understanding during these final year project development process.

Next, I would like to thank my project partner, Mr Koh Peng How for the cooperation to complete this project. Since this project are separated into two parts which is server side and client side. Each side are needed for another to complete the project. Without his assistants and cooperation, this project would not have been completed on time.

Moreover, I would like take this chance to thank all my friends for their contribution. Their opinions and supports to this project are important for me to further enhance it and produce a higher quality project. They also provided me with some interesting ideas to develop this project.

Lastly, I would like to thank my parents and my family for their supports during the process. Without their support and encouragement, this project would not be completed.

ABSTRACT

Nowadays, everyone is using at least one smartphone to assist them and to solve their daily problems such as daily scheduling, writing memo and managing their business. Hence, society is now increasingly interested in advanced mobile applications that will be on the user's advantage in solving their daily encountered problems. In addition, computer vision technology can be implemented in mobile applications since smartphones now are equipped with high resolution camera with high speed internet network.

In this project, a smart navigation system is designed to provide smart guidance to the smartphone users. The smart navigation system helps users retrieving information regarding a building or landmark such as the building name and the location of an office inside the building. The development of the proposed system is split into two parts, the mobile navigation client development and the building recognition server development. This report focuses on the development of the building recognition server.

First, the building recognition system will receive a picture and the coordinate information of the location where the picture is taken from the user's mobile phone. Second, the system will apply image processing technique to obtain the keypoints and their descriptors from the image. Next, the system uses the user's coordinate to retrieve buildings nearby the user's location pre-stored in the server database and their associated keypoint descriptors. Keypoint descriptors obtained from the input image will then be matched to the descriptors of each selected building. If successfully matched, the system will retrieve the information such as locations of offices inside the matched building from the database and send back to the user's mobile device.

Lastly, the building information sent from server will be processed by the mobile navigation system in order to display the information on the smartphone. The retrieved building information will be superposed on the picture and displayed in real time using augmented reality technology.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AR	Augmented Reality
BPBR	Biologically Plausible Building Recognition
BRIEF	Binary Robust Independent Elementary Features
DLA	Discriminative Locality Alignment
ERD	Entity Relational Database
FAST	Features from Accelerated Segment Text
GPS	Global Positioning System
HBR	Hierarchical Building Recognition
HTTP	Hypertext Transfer Protocol
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
LDA	Latent Dirichlet Allocation
LPP	Locality Preserving Projections
OpenCV	Open Source Computer Vision
ORB	Oriented FAST and Rotated BRIEF
PCA	Principle component Analysis
RANSAC	Random Sample Consensus
RDBMS	Relational Database Management System
SBID	Sheffield Building Image Dataset
SFBR	Steerable Filter-based Building Recognition
SIFT	Scale-Invariant Feature Transform
SURF	Speeded Up Robust Features
TCP/IP	Transmission Control Protocol/Internet Protocol
XML	Extensible Markup Language

CHAPTER 1 INTRODUCTION

1-1 Problem Statement

Although there are many types of navigation gadgets existing nowadays either in a device form or a mobile application installed in the smartphone such as Waze and Papago, which navigates the users to their destination and provide information about the road's condition. However, these navigation systems only guide the users to reach a destination such as a building but they do not provide additional information of the building or allow users to search for more information in a building. It can be a problem for the users upon reaching their destination if there are no sign boards or directories to the office around the building. They might get lost when finding their way inside the building thus cause a lot of inconveniences to the users.

Another scenario is that people might also want to know more about an unknown landmark or building located in front of them. For example, there would be a situation where a tourist wants to know what is the building in front of them and what is inside the building when they pass by some tourism places. Thus, a system that can provide building recognition capability and precise and informative navigation is important and very helpful to many users.

1-2 Motivation

One of the shortcomings of existing navigation systems is that they do not provide additional information regarding a destination to the users. For instance, the users are not able to find out the office location in a building unless they refer to the building directory. Therefore, an accurate, informative and enhanced navigation system is needed. The system should be able to provide the information about a building or a landmark to the users so that they can use that information to get around inside it.

To design this navigation system, computer vision and phone camera are needed to perform building recognition. The system also needs to collect the coordinate information of the users retrieved from satellite in order to restrict the number of image matching so that matching to improve matching efficiency and accuracy. In addition, augmented reality technique will be used to show building information by overlaying the information on the image.

1-3 Objectives

This project is aimed to build a smart navigation system which is able to perform building recognition and provide accurate information to the users. This system will be implemented on the server and provide service to the client side. The objectives of this project are:

- To develop a system that can recognize a building by using image and coordinate from client side and an image database
- To develop a system that can provide building information accurately to the client side

1-4 Proposed Approach

A Smart Navigation System will be developed to provide users a mobile navigation system with combination of building recognition system. It consists of two parts which are mobile navigation client and building recognition server.

For mobile navigation client, it focuses on mobile application which presents a platform to users to use the function of the system. The mobile application will be designed via android platform which will provide a user friendly interface to the users. It will cooperate with server side by transferring the essential data such as the location information and also the captured building image to the server for processing.

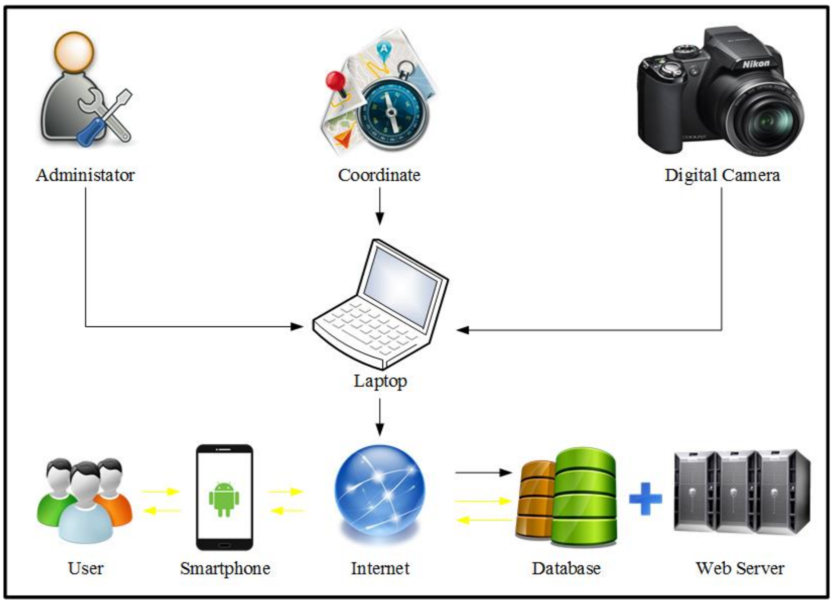


Figure 1-1: Smart Navigation System Design Diagram

In the building recognition server, it is separated into two phases which are learning phase and recognition phase. Before performing any recognition process, the building recognition system will undergo a learning phase where keypoint descriptors of all target buildings are pre-computed along with their building information as well as respective coordinates and the resulting data are stored into the database.

During recognition phase the system will focus on building recognition using the data retrieved from the client side which are captured building image and client's location coordinate. After that, it will use the location coordinate to filter the data in database before carrying out the recognition process. The system will be using Open Source Computer Vision (OpenCV), an open source computer vision library to perform the matching between the captured building image and filtered data. After successfully recognized a building, the system will send the details of the building such as the building name, offices and others information inside the building back to the client side.

1-5 Achievements

In this project, a smart navigation system will be introduced to provide users an accurate and efficient navigation system within UTAR campus area. This system will use building recognition technique to improve the accuracy of the navigation result. It also provides more information about the destination which is not provided by navigation system on market. The system will be divided into two parts which are mobile navigation system and building recognition system. Both of the systems will need to collaborate with each other to perform the building recognition and to provide navigation to the users.

The system implements building recognition technique to improve the accuracy of the result while the augmented reality technique is used to provide a more user friendly result to the users. The separation of the system is to ensure the process of navigation able to response in real time. At last, users will be able to know the specific location of their destination by just taking a picture of the building using their smartphone which equipped with camera and Global Positioning System (GPS). Therefore, it can save users' time and able to provide an enhanced version of navigation system to the users.

1-6 Report Organization

The report are divided into 6 chapters. In chapter 1, the problem statement and motivation of the project had been stated clearly. Chapter 1 also stated what have been achieved in this project. While, in chapter 2, six literature review had been reviewed in order to provide idea to the project development process. By referring to the literature review, there had some techniques been implemented in the project.

In addition, the design of the system had been explained clearly in chapter 3. This chapter will provide the system overview and allow reader able to have a clear understanding about the system. Furthermore, methodology and the tools used will be explain in chapter 4. It will explain how the system has been developed and the specification of hardware and software for the system. Next, an implementation and testing result of the system had been carried out and it is explained in chapter 5. A few testing had been conducted to evaluate the performance of the system.

Lastly, the limitation of the system and future works may implement in the system will be stated in chapter 6. Besides that, there will be a conclusion in this chapter which will concludes what had achieved in this project.

1-7 Background Information

The built-in GPS function on the smartphone is no longer a new technology in this era however it plays an important role to the phone users in their daily life. It is able to guide phone users to their destination with the help of the coordinates retrieved from satellite. The coordinates filter the required data that used for recognition to increase the efficiency of the system.

The OpenCV is an open source library provided different type of function in image processing. The proposed building recognition system will use functions in the OpenCV library to find out the keypoint and keypoint descriptor which are those point that can represent the object in the image. After that, it will use building recognition technique to perform matching on the keypoint descriptor and data in the database.

The PHP is a server side scripting language that used to develop the web system in this project. The PHP language provide a lot of function which can easily implement in the system. A C++ program able to be execute by using PHP scripting and the result will return back to the script and undergo further processing. It able to identify all the error when the scripting unable to process.

A PHP Socket was used in communicate between the client and server in transmission of the data. The socket is used the Transmission Control Protocol/Internet protocol

(TCP/IP) to transmit the data between two parties. In this project, the system uses the socket as the middle men to transmit the image data and coordinate from the mobile to the server to process it. After finish processing, the server will send the result back to the mobile using this socket. The socket will send according to the IP address and port number.

After successfully recognize the building, the building information will be retrieved from database and send it back to navigation system. The result will be the information about the building such as name of the building and its contact information. Next, the navigation system will process the result and display it using augmented reality technique. It will overlay the information of the building on the real time image and users will be able to understand the information easily.

CHAPTER 2 LITERATURE REVIEW

2-1 Literature Review

In urban areas, people tend to determine the desired destination based on their knowledge about the current location. Furthermore, more location information is able to obtain through the use of GPS while the image of the landmarks respect to the person also will give some additional details for the location. Hence, it is necessary to have a stable and accurate landmarks recognition to enable such functionality (Zhang & Kosecka, 2005).

Commonly, there are a few necessary steps in order to perform recognition such as feature point extraction and finding descriptor. Besides that, the amount of data transferred between server and client side is also an important issue because it may delay the data receive time and output incorrect data to the users. In order to solve the listed problems, a lot of models and techniques will be introduced and discussed here.

2-2 Recognizing Building Based on Local Oriented Features

There are a lot of building recognition systems had been proposed recently. Nonetheless, most of the system are used to recognize the building which depends on a complex feature extraction procedure. Since every retrieved image of the building contains different amount of variability such as different viewpoints, different lighting conditions and occlusions, therefore the building recognition system must be able to solve the problem mentioned above in an efficient way.

Therefore, a building recognition model which focuses on local oriented features had been presented by Li and Allinson (2013). This model are in modular, simple and computationally efficient. The building recognition model is named as Steerable Filter-based Building Recognition (SFBR) model. It is distributed into 3 important modules: feature representation, feature pooling, and reduction of dimensionality as shown in Figure 2-1.

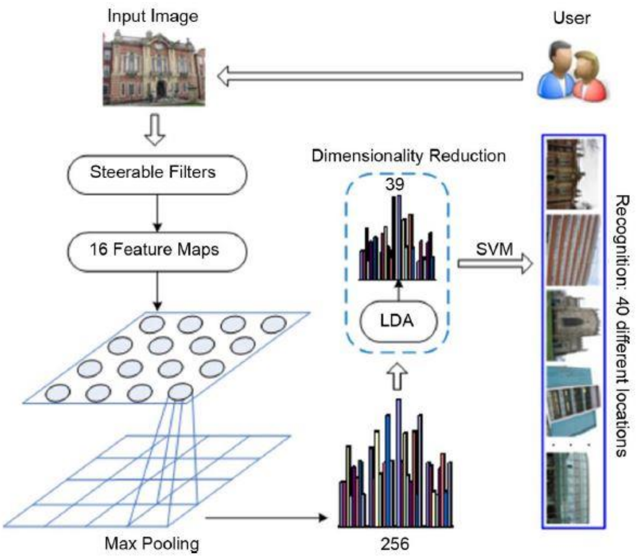


Figure 2-1: Steerable Filter-based Building Recognition (SFBR) (Li & Allinson, 2013)

2-2-1 Feature Representation

A calculation of steerable filter depends on various orientation will produce the feature representation. Steerable filter is a combination of several filters with arbitrary orientation. Next, the model will use second-order Gaussian and it corresponds to Hilbert transform.

The second order filter will produce a result which is calculated at eight different orientations. For each of the image, the result will output a total of 16 feature maps. The reason to choose second order steerable filter is because it uses 7 basic functions of Gaussian and Hilbert transform to shift to arbitrary orientation. Besides that, the order of steerable filter is directly proportional to the computational cost. When the order of steerable filter increases, the computational cost increases as well.

2-2-2 Feature Pooling

Pooling is often used in image recognition algorithms because it can generate position-invariant response. There are two type of pooling which are sum pooling and max pooling. Since sum pooling assigns equal weight to input and it will causes some lost in feature specificity, hence it is not able to resolve the size invariance problem. Therefore, max pooling had been chosen in this model because it can obtain more complex representation without affected by the image noise.

The max pooling obtains the complex representation by finding the maximum value of the steerable responses from local patches. It can prevent from losing information while ignore those irrelevant feature responses and transform into an accurate representation. In this model, there is a need to adjust the max pooling step carefully to achieve a good result. Based on Figure 2-1, each feature map is equally distributed into 16 regions and every image will be transformed into a 256-dimensional feature vector.

2-2-3 Dimensionality reduction

In dimensionality reduction, it will use Latent Dirichlet Allocation (LDA) to preserve the discriminative information that will be used in recognition process later. LDA is a supervised learning algorithm and it will convert the dimension of feature vector from 256 to 39. It is used to moderate the computational complexity and it also will make sure that there is enough discriminative information for the following recognition process.

2-2-4 Experiments and Evaluation

A comparison of the result between SFBR and another two building recognition systems which are Hierarchical Building Recognition (HBR) system and Biologically Plausible Building Recognition (BPBR) scheme. The results of comparison are shown in Table 2-1 which shows that SFBR give a good result on accuracy with 94.66%.

Table 2-1: Comparison Result of Building Recognition Algorithms (Li & Allinson, 2013)

Performance Evaluation	**Building Recognition Algorithms**		
	HBR	**BPBR**	**SFBR**
Average Precision (%)	73.32%	85.25%	94.66%

In addition, an evaluation on the different ways to reduce the dimension in SFBR had been conducted. The LDA will be compared with another two algorithms which are Locality Preserving Projections (LPP) and Discriminative Locality Alignment (DLA). The result of comparison are shown in Table 2-2 which shows that LDA has higher performance compare with others.

Table 2-2: Comparison Result of Dimensionality Reduction Algorithms (Li & Allinson, 2013)

Performance Evaluation	Dimensionality Reduction Algorithms in SFBR		
	DLA	LPP	LDA
Average Precision (%)	88.43%	93.39%	94.66%

2-3 Client-Server Architecture of Mobile Augmented Reality

An image recognition system using mobile augmented reality (AR) had been proposed by Gammeter, et al. (2010). This approach is a hybrid approach which is distributed into 2 sides which are server side and client side as shown in Figure 2-2. It provides a lot of benefits in object recognition as stated below.

- It is able to obtain objects details from a huge database within short time without storing any database on client side
- It can send real time image to the server without clicking it manually to process
- It minimizes the usage of communication between client side and server side

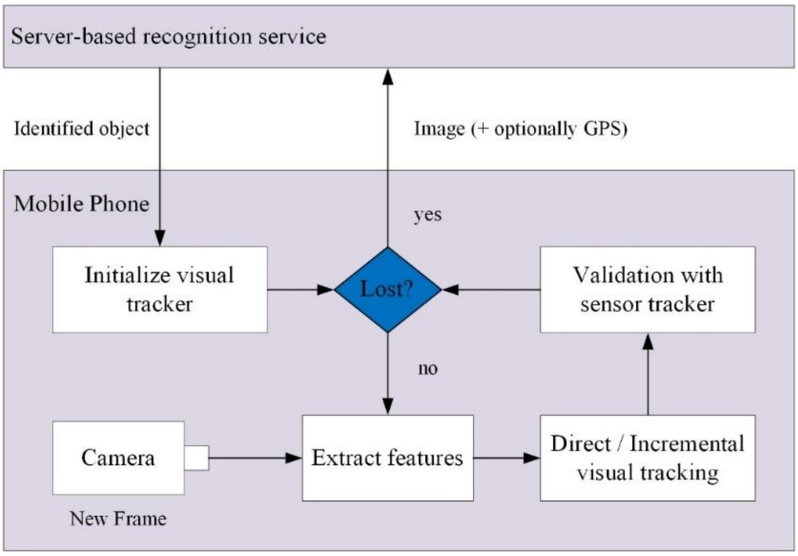


Figure 2-2: Overview of hybrid approach (Gammeter, et al. 2010)

2-3-1 System overview

The proposed system basically divide the recognition and tracking operations into two sides which are server side and client side. The server side will handle recognition process while the client side will use augmented reality technique to perform tracking operations. On client side, it will ask for recognition to the server side by transmitting the capture real time image along with the coordinate through a Hypertext Transfer Protocol (HTTP) connection. While, the server side will be reply an Extensible Markup Language (XML) data which contains the location information such as title and the bounding box coordinate. It will be used by client side to start the tracker and label augmentation on the phone.

2-3-2 Server side object recognition

In server side, server will receive the image and coordinate from client side and recognize the object in the image. The recognition process need a scalable, efficiency retrieval method and an image database. The database should contain the information about the building or landmark such as the building name. In order to have a sufficient amount of image to perform recognition process, server side will crawl geotagged images from Flickr and group the same location by matching their similarity using Speeded Up Robust Features (SURF) as shown in Figure 2-3. The product of this grouping will be a set of clusters which will be labelled and record the Meta data into the database. Moreover, each images will calculate the bounding box of the object's position.

Figure 2-3: A group of same landmark (Gammeter, et al. 2010)

Next, the images in the cluster will be categorized by using approximate k-means for visual words learning. After that, all query image will undergo the same procedure as stated previously and perform matching by using the 1 million of visual words. Since computational cost will be too high, hence it uses inverted file structure to perform matching. In order to maintain the consistency, this approach will use 500 potential images to perform geometric checking by using Random Sample Consensus (RANSAC) estimation. The output will be the matching score for this 500 images.

2-3-3 Client side object tracking

After the server side send back the result of recognition, a virtual label will be augmented on the mobile screen with respective coordinate. Next, it will perform several steps which are visual feature tracking, motion estimation and sensor tracking in order to track the object. It will use Features from Accelerated Segment Text (FAST) corners for tracking and used the saved features in users' smartphone as a reference.

When the correspondences of features are found, it will perform motion estimation. After that, the system will combine the visual tracking with sensor tracking. As a way to overlay a label accurately on the real time image, a lot of filtering need to be performed because the devices will produce some noise signals and the sensor unable to provide a consistent time intervals. When object tracking is failed, it will request the server side to perform the recognition process again to track back the object.

2-4 Scale Invariant Feature Transform (SIFT)

An enhanced version of Scale Invariant Feature Transform (SIFT) algorithm had been proposed by Lowe (2004), the origin inventor of SIFT. The SIFT can be used to perform dependable matching in various situation. The advantages of SIFT algorithm are:

a) Invariant to image scale and rotation

b) Robust to different image quality such as noise

c) Robust to 3D viewpoint

Therefore, the SIFT algorithm is able to detect the object image accurately without affected by other issue such as noise. There are four stages in calculating the SIFT keypoint and descriptors where it perform matching between the two different images. The four major stages of this algorithm are stated below:

a) Scale-Space Extrema Detection

The keypoint will be obtained by using the difference-of-Gaussian function where it will find those interested point from different views of the same object. In order to make the algorithm faster, scale space function is implemented and it is defined as:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$

where $G(x, y, \sigma)$ represents variable-scale Gaussian, * represents convolution operator and $I(x, y)$ represents input image. Gaussians function had been used to find out the stable keypoint from scale space where it computes the difference between an image and another identical image but with different. The Gaussians function is defined as:

$$D(x, y, \sigma) = \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y)$$

$$= L(x, y, k\sigma) - L(x, y, \sigma)$$

Next, it will use each point to compare with eight neighbours in the identical scale image and nine neighbours for the top-scale and low-scale to obtain the local maxima and minima of $D(x, y, \sigma)$ as shown in Figure 2-4. When the result is the minimum or maximum among others, then it will be the extrema and it is indicated as the SIFT keypoint.

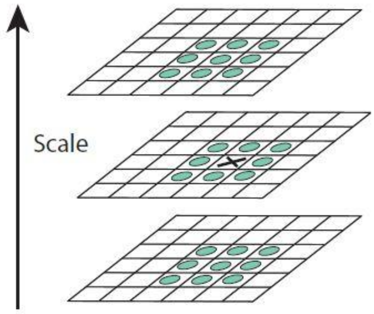


Figure 2-4: Maxima and minima of the difference-of-Gaussian images (Lowe, 2004)

b) Keypoint Localization

It is an important step to ensure the accuracy of image feature matching and increment of the anti-noise ability is needed. It will start to remove the low-contrast and unstable keypoints by calculating the Laplacian value for every keypoint found in extrema detection.

c) Orientation Assignment

In this stages, the purpose is to allocate a reliable orientation to the keypoint by using the local image properties to ensure it is invariant to rotation. It implements two equations in order to find out the gradient magnitude, m and orientation,μ as stated below:

$$m(x,y) = \sqrt{(P(x+1,y) - P(x-1,y))^2 + (P(x,y+1) - P(x,y-1))^2}$$

$$\mu(x,y) = tan^{-1}(\frac{P(x,y+1) - P(x,y-1)}{P(x+1,y) - P(x-1,y)})$$

where P is Gaussian smoothed image.

d) Keypoint Descriptor Computation

The gradient found from previous stage will be transformed into keypoint descriptors. First, the variance of 1.5 times the keypoint scale is set and the Gaussian will compute the gradient. Next, a collection of 16 histograms will be arranged in a 4x4 array and assign 8 orientation bins to each of them. Thus, the result will be 128 element feature vector for each keypoint.

2-5 Performance Evaluation of SIFT-Based Descriptors for Object Recognition

An evaluation on SIFT has been conducted by Tao, et al. (2010) in order to find out the pre-request for object recognition and the performances of the SIFT descriptor with enhanced version SIFT descriptor. The standard stages for an object recognition are feature extraction and feature matching. It will search all the extrema points and from those points obtain the position, scale, rotation invariant feature vector. While for the SIFT descriptors, it is separated into 4 parts:

a) Detect scale-space extrema

b) Locate keypoints

c) Assign orientation

d) Compute keypoint descriptors

2-5-1 Development and Improvement of the SIFT Descriptor

Two main focuses had been proposed in order to improve the SIFT Descriptor performances. First, it uses grayscale images to find the keypoint descriptor by using different histograms, different region shapes or reduce the dimensionality. Second, it suggests to use images in colour space such as HSV or RGB.

2-5-2 Performance Evaluation of SIFT Descriptors for Object Recognition

The performances of SIFT descriptor will be based on a suitable representation of the image and the efficiency of image matching and recognition. It is stated that SIFT descriptors have a lot of good properties and it invariant to:

a) Image condition such as translation, rotation, reduction or amplification

b) Environment condition such as brightness, occlusion and noise

The SIFT features is fast and optimized and thus the matching process can be achieved in real time.

2-6 Speeded-Up Robust Features (SURF)

A scale and rotation invariant recognition algorithm had been proposed by Bay, et al. (2007) named as Speed-Up Robust Features (SURF). It is targeted to find a balance between computation cost and performance such as simplify the detection but still produce a high accuracy result.

2-6-1 SURF Detector

There are 4 steps to find out the feature point by using SURF detector. First, it will use integral image to represent the image. By using integral image, it can compute any size of rectangle region as shown in Figure 2-5 with high speed. The integral image is the total of all pixels within a rectangle region in the input image as shown in the following equation.

$$I_{\Sigma}(x) = \sum_{i=0}^{i \le x} \sum_{j=0}^{j \le y} I(i,j)$$

Where $I_{\Sigma}(x)$ is an integral image at a location *x=(x,y).*

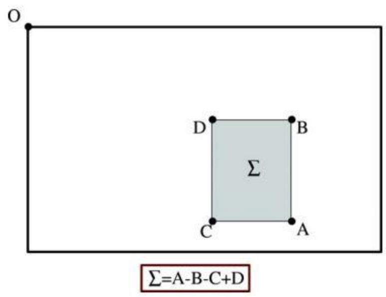


Figure 2-5: Integral Image (Bay, et al., 2008)

Next, it will use Hessian matrix as the detector due to its high accuracy. It can detect blob-like structures although the determinant of the location is highest. The Hessian matrix are declared as the following equation.

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

Where $H(x,\sigma)$ is a Hessian matrix in a point at x with σ scale and the $L_{xx}(x,\sigma)$ is the product of filter smoothing of the convolution of the Gaussian second order derivative and same with another three.

Furthermore, it will implement scale-space as the image pyramid which can scale up the Gaussian filter size without reducing the image size. It will use 9x9 filter as the initial scale layer and 1.2 as the scale. Moreover, there will be a feature point localization which implements a non-maximum suppression in a 3x3x3 neighbourhood to locate the feature points in the image.

2-6-2 SURF Descriptor

There are two steps in the descriptor finding process where are orientation assignment and Haar wavelet responses. First, it will take the image to convolute with two first-order Haar wavelets. The result will be output in a two dimensional space. After that, it will use a window of size $\frac{\pi}{3}$ to calculate the summation of the result and the longest resulting vector will be the orientation. Next, a 4x4 square sub-regions will be built on every feature point by referring to the selected orientation. Lastly, it will produce a descriptor vector and it will be extracted to become SURF descriptor.

2-7 ORB: an efficient alternative to SIFT or SURF

An object recognition method has been proposed by Rublee, et al. (2011) named Oriented FAST and Rotated BRIEF (ORB) which is also known as combination of

FAST keypoint detector and Binary Robust Independent Elementary Features (BRIEF) descriptor. ORB is an enhanced version of the combination of FAST and BRIEF which deals with rotation invariant and free from image noise problem. It is suitable for real time performance and low-power devices.

2-7-1 oFAST: FAST Keypoint Orientation

FAST has been chosen to find keypoint in ORB due to its high computational efficiency. However, it is still unable to solve the orientation problem. Hence, an enhanced version of FAST named oFAST has been proposed in order to solve the mentioned problem. First, this approach will start to attain the intensity threshold between the centre pixel and the ones within the radius of 9. This approach chooses FAST-9 with a circular radius of 9 to obtain the intensity due to its good performance.

Moreover, FAST do not have a corner measure and it just focuses on edges. So, oFAST implements a Harris corner measures to arrange the keypoint. Next, oFAST also implements a scale pyramid to solve the scale invariance problem. The scale pyramid is formed by producing filtered FAST features at each level in the pyramid.

Next, ORB using intensity centroid to measure corner orientation although it is simple but it is effective to deal with corner orientation. Firstly, it will find the moments and the moments of a patch are stated as:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y)$$

The moment will be used to find out the centroid as stated below:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right)$$

After that, from the corner's centre, 0, a vector will be built to the centroid. The patch orientation is $\theta = atan2(m_{01}, m_{10})$ where *atan2* is the quadrant-aware version of *actan*. In order to improve the rotation invariance, the moments must compute with the remaining x and y inside a circle of radius r. This centroid will output a stable orientation although it is under a serious image noise.

2-7-2 rBRIEF: Rotation-Aware Brief

ORB had proposed a enhance version of BRIEF which named as steered BRIEF descriptor. Even though has only a slight difference of degree, the BRIEF performance will be degraded when performing matching on in-plane rotation. Thus it will use steered BRIEF method based on the orientation of the keypoints.

First, a 2 x n matrix will be produced by performing n binary tests at location(x_i, y_i). Next, it will use the patch orientation θ and the respective rotation matrix R_θ to build an enhancement of S, $S_\theta = R_\theta S$. After that, it will discretize the angle to increment of 12 degrees and build up a lookup table of pre-computed BRIEF patterns.

However, the problem of using steered BRIEF is it will lose some variances and affect the correlation when performing binary test. Hence, a strategy to tackle this problem is to apply Principle Component Analysis (PCA). After finish searching all possible binary tests, a vector T will be obtained by ordering the test result from a means of 0.5. Next, a series of uncorrelated tests with an approximately means of 0.5 will be conduct by using greedy search. It will show an enhancement of variance and correlation over steered BRIEF and the result named as rBRIEF.

2-8 Critical Remarks of previous works

There are several methods that can be used to resolve the problem in this project. However, there are a few disadvantages in each approach that had been reviewed.

First, the reviewed approach that uses local oriented features to perform building recognition as mention in section 2-2 consists of a few disadvantages throughout the whole recognition process. It may consume high amount of resource when performing building recognition in the server. It wasted a lot of resources when it retrieves or saves the image into the database rather than just saving those information or image data into the database. Besides that, it may need to do some data filtering such as using the coordinate of the users to filter the data retrieved from the database in order to minimize the consuming of server resources. Hence, it will increase the efficiency and accuracy of the system so that it is able to verify the data whether the location is correct or not by using the coordinate.

Second of all, the reviewed system approach that separate into server side and client side as mention in section 2-3 provides a basic idea to design the smart guide system in this project. However, it still consists some minor problem which may affect the result of recognition. The sensor built on the smart phone may produce noise signals and it will results in irregular time intervals. Therefore, it is a need to undergo some filtering on it before the tracking starts. Besides that, it needs to use 500 images to perform checking but it is resource consuming. Hence, it need some data filtering with the reference of the users coordinate.

Thirdly, another recognition algorithm, SIFT had been reviewed in section 2-4 where it provides an object recognition algorithm. It is able to recognize the object although it is different in scale, captured from different view or with some image quality issue such as noise. The processing time for SIFT is quite fast compared to other algorithm however the ORB algorithm is still faster than SIFT. Besides that, SIFT will output a lot of keypoint descriptor and it may affect the computational cost.

Fourthly, a SIFT based descriptors evaluation had been reviewed as mention in section 2-5 which shows the performance of the SIFT on image matching and recognition. It shows that SIFT is suitable to use as an image recognition algorithm because it contains a lot of advantages such as scale and rotation invariant and robust to noise. It is one of the important factors in recognition process and it may influence the result. However, there are some disadvantages such as the complexity of the descriptor. It can be improved by using the proposed technique such as using different histograms and different region shapes.

Fifthly, another detector and descriptor algorithm, SURF had been reviewed as mention in section 2-6 which shows the enhancement version of SIFT in terms of speed and performance. It focuses on the balancing of performance and speed, but it is still not faster than ORB. Since this project focuses on building recognition, therefore an algorithm which is good in scale and rotation invariance is needed. However, SURF only provides a common scale and rotation invariance while SIFT is better in scale and rotation invariance compared to SURF.

Lastly, a recognition algorithm, ORB had been reviewed as mention in section 2-7 will provide simple function to perform object recognition but it still consists of some problem when using it. ORB is not able to solve the scale invariance problem although it uses the scale pyramid by produce the feature point on each level of the pyramid. Hence, there is a need of future work to improve this algorithm.

For this building recognition system, it will use some techniques from the reviewed approach in order to complete the smart navigation system. The architecture of the system will be referred to the approach that is mentioned in section 2-3 which divides the system into server side and client side. It will be easy and efficient to perform such high computation cost process. On the other hand, for the recognition process, the system will use the SIFT algorithm to perform recognition as it is a stable recognize algorithm. Although ORB shows that it is faster than SIFT, it still not able to provide more accurate result compared to SIFT. Furthermore, it easy obtained from OpenCV library and it is more efficient compared to other algorithms such as SURF.

CHAPTER 3 SYSTEM DESIGN

3-1 System Design

In this building recognition process, it contains 2 phases which are learning phase and recognition phase. The main purpose of the learning phase is to input the images of buildings to be recognized, coordinates of the buildings, specification of the images and the building information into database. The data are used for recognition process later during recognition phase. On the other hand, the recognition phase shall recognize the input image from client side using the data in database. During recognition phase, coordinate of the input image is used to perform data filtering in the database and the selected data will be used to perform matching with the input image. The building recognition server development is as shown in Figure 3-1.

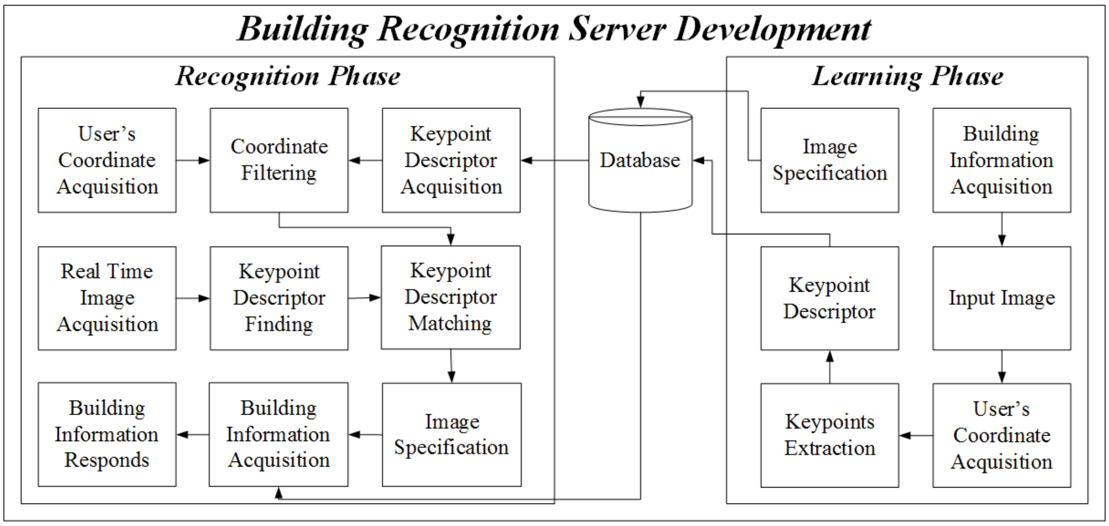


Figure 3-1: Building Recognition Server Development Block Diagram

3-1-1 Learning Phase

a) Building Information Acquisition

The building information will be obtained by referring to some external sources such as website and map. The building information contains the name of the building, the faculty or department in the building, the office phone number, and the website of the department as shown in Figure 3-2.

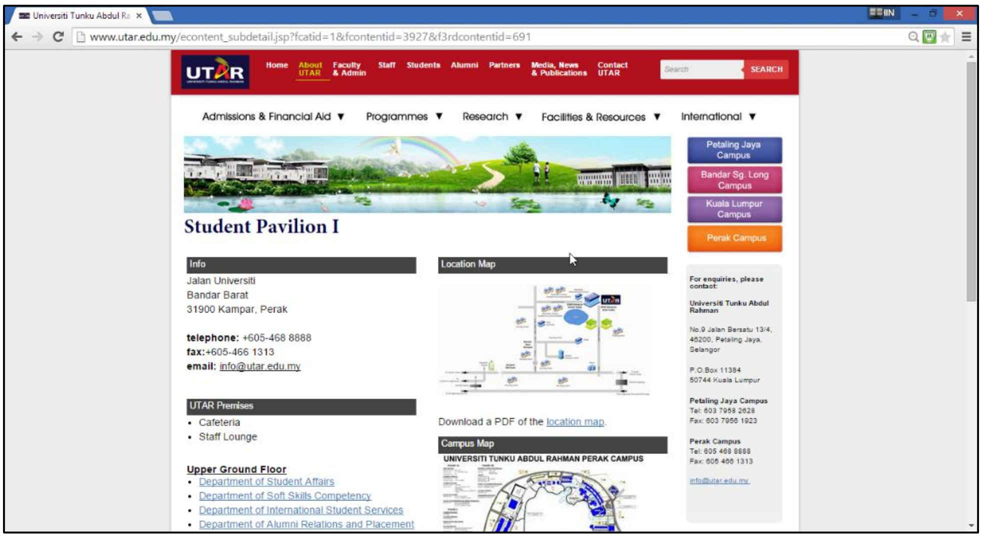


Figure 3-2: Sample of external website

After successfully obtain all the needed information, the building name is input into the database by submitting a new building form through the website as shown in Figure 3-3. An updated building name will show on the table located at the bottom of the website.

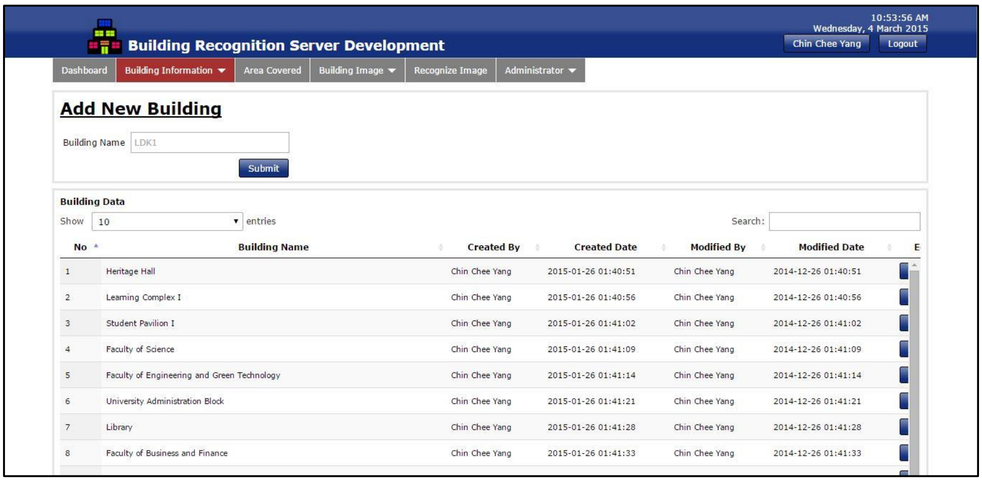


Figure 3-3: Add New Building Form

The next step will be input all the building information such as department name, phone number, email and the others into the database. These information will be fill up into the new building information form and submit as shown in Figure 3-4. After successful input, the respective updated building information will show on the table located at the bottom of the website.

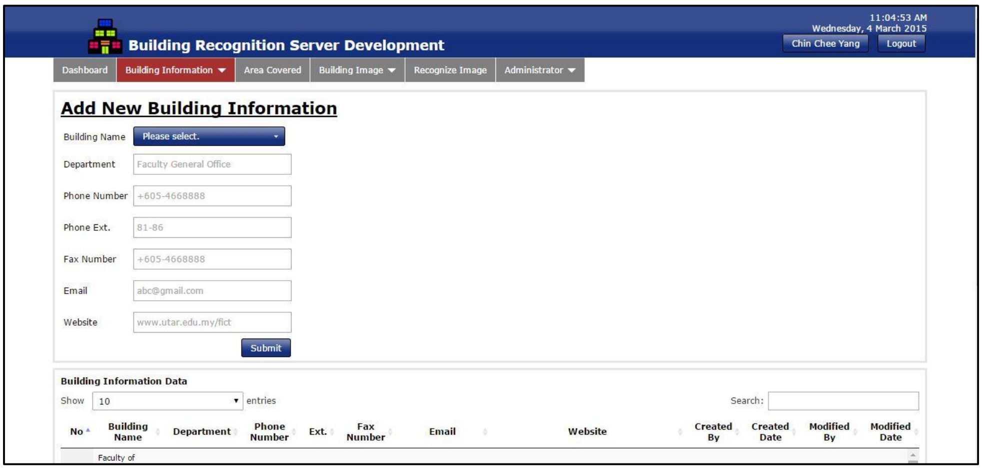


Figure 3-4: Add New Building Information Form

b) Input Image

The building image will be taken by using digital camera. The input image should completely cover each of the building sides as shown in the Figure 3-5. Since the size of the captured images might vary, it will undergo a resizing process that will resize all images to standard dimension of 640 pixels x 480 pixels. After resizing, the image will be saved as Portable Network Graphic (PNG) file format which are having lossless compression advantages.

Figure 3-5: Sample Input Images

There are few types of image file compressed format such as Joint Photographic Experts Group (JPEG) and PNG. However, JPEG is a lossy compression which means that the reproduction of the original after decompression does not guarantee the quality of the images. Therefore, in this project all input image are saved in PNG file format.

After successful converting the image into required pixel dimension and file format, upload the image under the Add New Building Image form as shown in Figure 3-6.

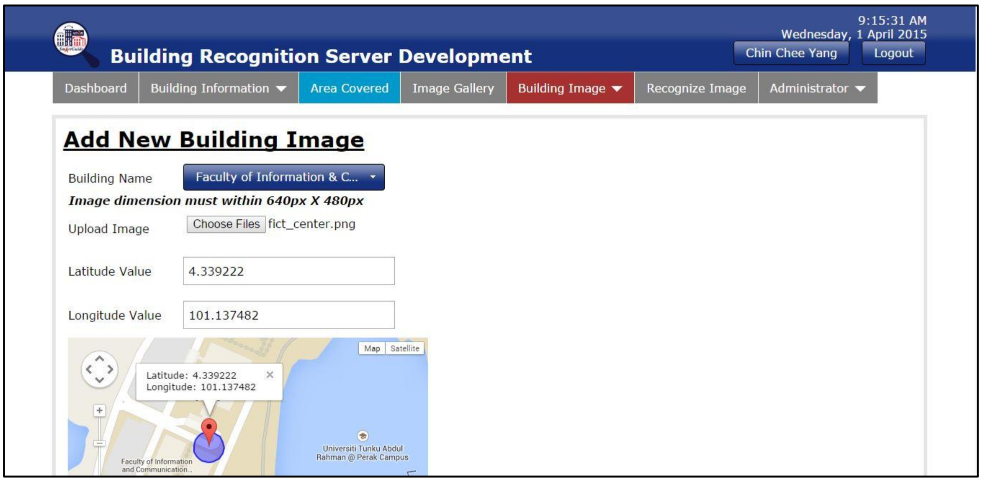


Figure 3-6: Add New Building Image Form

c) Users's Coordinate Acquisition

After selecting the image, it needs the coordinates of the location of the photo taken. In the Add New Building Image form, there are two input boxes to input the latitude and longitude value. The coordinate value can be determined by referring to the Google Map section located below the input box. The coordinates are in decimal degrees form such as (4.339222, 101.137482) where the 4.339222 is latitude value and the 101.137482 is longitude value.

With the aid of the Google Map, all user need to do is to click on the map to select the location of the photo taken. When the user click on the Google Map, there will be a marker indicated the location of the users clicked as shown in the Figure 3-7. It will auto update the value into the input box.

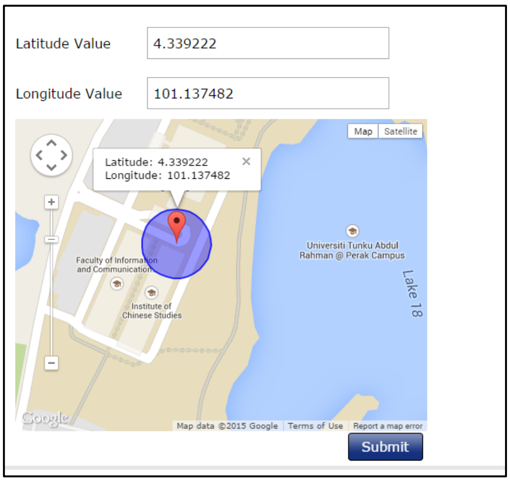


Figure 3-7: Building Coordinate in Google Map

After finish selecting all the required option, the user may click the Submit button to upload the image and other information into the server. The coordinates will be saved into the database along with the building id respectively. The uploaded image will undergo some image processing processes as explained in the next section.

d) Keypoints Extraction

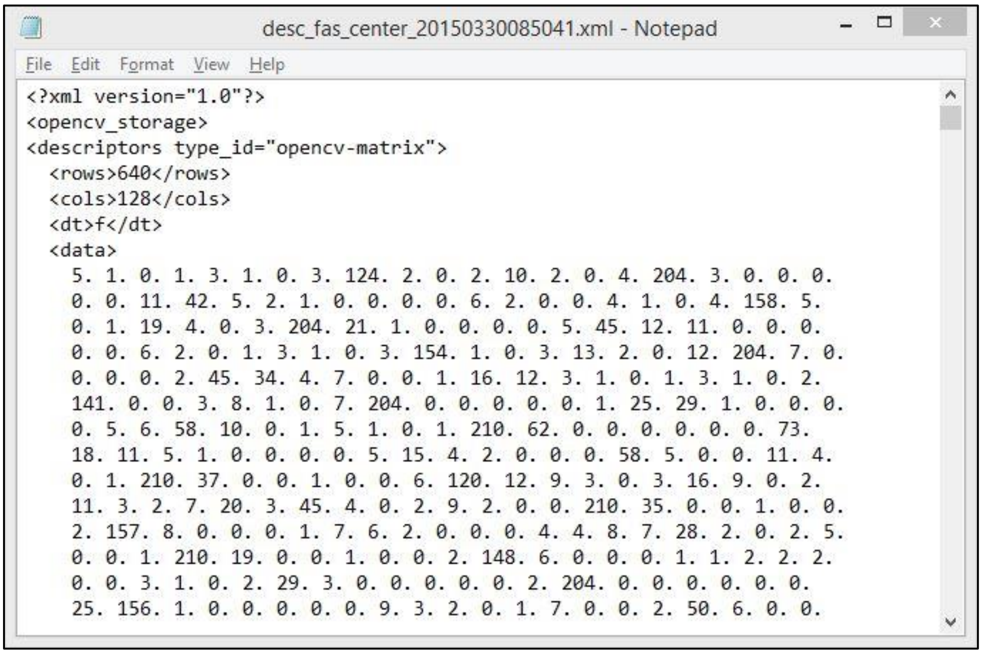
After the image has been successfully uploaded to the server, the server will execute a C++ program to process the uploaded image. The system uses SIFT to extract feature keypoints on the image. SIFT is an algorithm that can be used to detect and describe the feature points on an image. There will be few steps required to follow in order to find out the keypoints on the image.

First, the program will load the input image into a grayscale image using a method provided by OpenCV. It is because it can simplify the processing and the colour of the image did not provide a significant effect on the result. After that, the grayscale image will duplicate and resize it using a provided resize method to half of its dimension. There will be two different dimension of grayscale image that needed to obtain the keypoints which are 640px X 480px and 320px * 240px.

Next, the system use the SIFT feature detector to detect all the keypoints on the two grayscale images. The detected keypoints are the interest points found in an image and it will be saved into a vector. After that, it will go through the keypoint descriptor to find out the characteristics of the keypoint and it will be used for matching in the recognize phase.

e) Keypoint Descriptor

The keypoint descriptor extracts discriminative features from keypoints that are used to perform matching with other images. It will be obtained from the keypoint descriptor extractor function where is implemented into the system to find out the characteristic of an images. The function is inside the OpenCV library and it is easy to implement it on the system. After successfully extracted all the descriptor of an image, it will be saved as XML file into the system for future landmark recognition work as shown in Figure 3.8.


desc_fas_center_20150330085041.xml - Notepad

File Edit Format View Help


```
<?xml version="1.0"?>
<opencv_storage>
<descriptors type_id="opencv-matrix">
  <rows>640</rows>
  <cols>128</cols>
  <dt>f</dt>
  <data>
    5. 1. 0. 1. 3. 1. 0. 3. 124. 2. 0. 2. 10. 2. 0. 4. 204. 3. 0. 0. 0.
    0. 0. 11. 42. 5. 2. 1. 0. 0. 0. 0. 6. 2. 0. 0. 4. 1. 0. 4. 158. 5.
    0. 1. 19. 4. 0. 3. 204. 21. 1. 0. 0. 0. 0. 5. 45. 12. 11. 0. 0. 0.
    0. 0. 6. 2. 0. 1. 3. 1. 0. 3. 154. 1. 0. 3. 13. 2. 0. 12. 204. 7. 0.
    0. 0. 0. 2. 45. 34. 4. 7. 0. 0. 1. 16. 12. 3. 1. 0. 1. 3. 1. 0. 2.
    141. 0. 0. 3. 8. 1. 0. 7. 204. 0. 0. 0. 0. 1. 25. 29. 1. 0. 0. 0.
    0. 5. 6. 58. 10. 0. 1. 5. 1. 0. 1. 210. 62. 0. 0. 0. 0. 0. 0. 73.
    18. 11. 5. 1. 0. 0. 0. 0. 5. 15. 4. 2. 0. 0. 0. 58. 5. 0. 0. 11. 4.
    0. 1. 210. 37. 0. 0. 1. 0. 0. 6. 120. 12. 9. 3. 0. 3. 16. 9. 0. 2.
    11. 3. 2. 7. 20. 3. 45. 4. 0. 2. 9. 2. 0. 0. 210. 35. 0. 0. 1. 0. 0.
    2. 157. 8. 0. 0. 0. 1. 7. 6. 2. 0. 0. 0. 4. 4. 8. 7. 28. 2. 0. 2. 5.
    0. 0. 1. 210. 19. 0. 0. 1. 0. 0. 2. 148. 6. 0. 0. 0. 1. 1. 2. 2. 2.
    0. 0. 3. 1. 0. 2. 29. 3. 0. 0. 0. 0. 0. 2. 204. 0. 0. 0. 0. 0. 0.
    25. 156. 1. 0. 0. 0. 0. 0. 9. 3. 2. 0. 1. 7. 0. 0. 2. 50. 6. 0. 0.
```


Figure 3-8: Sample Trained Descriptor XML

f) Image Specification

The specification of the image such as the location of a room, location of an office on the building and others was inputted into the system by using Add New Image Specification Form as shown in Figure 3-9. There will be four corners which are corner A, corner B, corner C and corner D. Each of the corners indicates the corner of the respective room in the building. After clicking on the image, a shaded rectangle is shown to indicate the selected area for the room on the building.

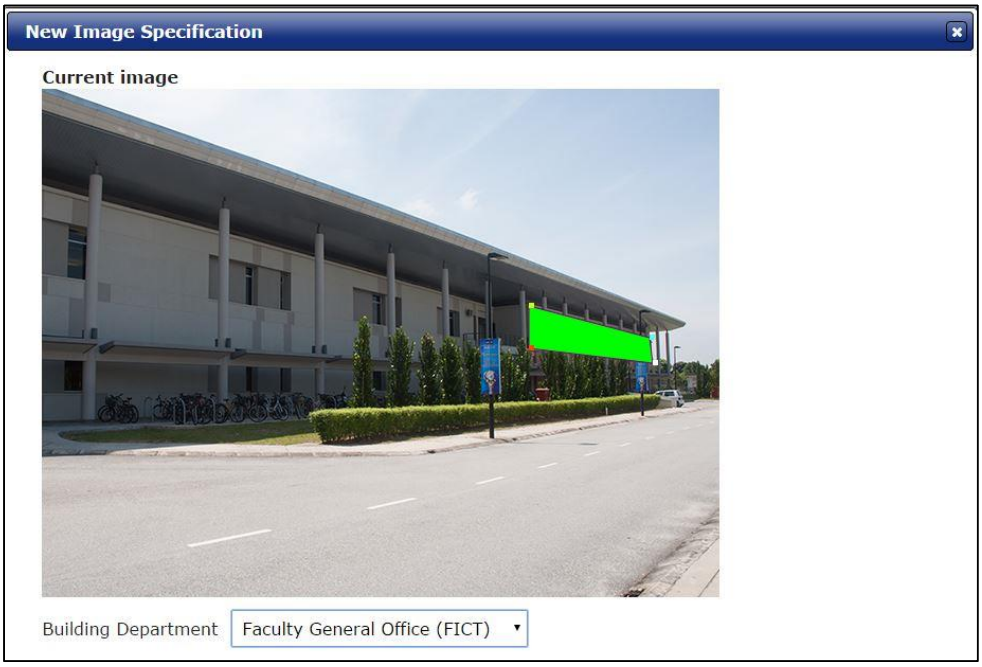


Figure 3-9: Image Specification Sample

After successfully select the 4 corner, the user can submit it and the 4 corners will be saved into the database. The system will used the saved corner to mark the room on the building when performing building recognition.

3-1-2 Recognition Phase

a) Users' Coordinate Acquisition

The mobile navigation system will obtain the users' current location coordinate by using the phone GPS and send to the building recognition server. After that, the mobile system will append the coordinate into a XML file with a latitude and longitude tag to indicate their respective location.

b) Real Time Image Acquisition

The real time image is captured by using the mobile phone camera and it will be saved in PNG file format. After that, the mobile navigation system converts the image to a grayscale image. Since the image file size is big and needs some time to transmit over the internet, thus the image is compressed by using the imencode function in the OpenCV. The result of the encoded buffer is in a vector format and saved it into an XML format with buffer tag. The image is a Matrix format which every pixel having its own value to indicate the intensity of that particular pixel. The value is in between 0 to 255.

c) Coordinate Filtering and Keypoint Descriptor Acquisition

After successfully receive the coordinate from the XML, the system uses it to filter the pre-stored keypoint descriptors of buildings in the database. The system retrieves keypoint descriptors that are within 50 meter of the filtered coordinates as shown in the Figure 3-10. This is to prevent retrieving unnecessary data from database that might affect system performance.

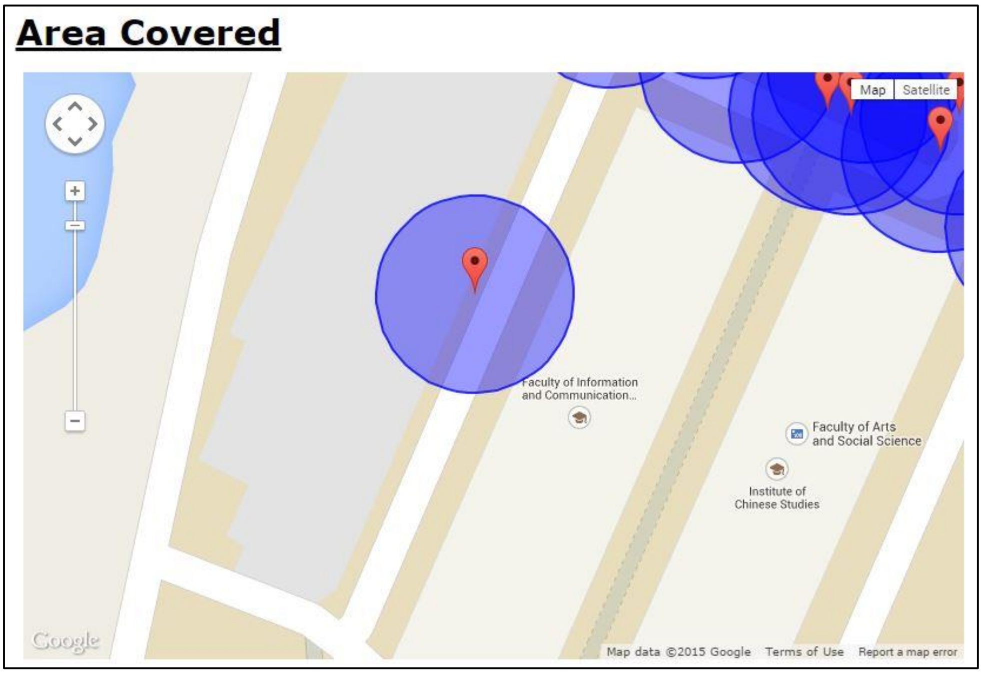


Figure 3-10: Coverage of a Coordinate

The database management system will return back the stored keypoint descriptor file name and it will save into a temporary text file which will input to the recognition program later. The related keypoint descriptor will be used for matching on the following process.

d) Keypoint Descriptor Finding

After all the needed information are saved into a temporary text file, the system execute the C++ program with the input of the temporary text file name and the path of the image. The C++ program will analysis the content of the text file and saved it into Vectors for processing later. After successful retrieved all the content, the system decoded the buffer into an image.

The program performs resizing on the image to duplicate another small dimension image. There will be two different dimension of images same as the training phase. The program extract the keypoints and find out all the descriptor from the two image by using the SIFT algorithm.

e) Keypoint Descriptor Matching

There are two stages in the keypoint descriptor matching. First, the system are uses the small dimension of image to perform matching. The reason of using the small size of image is to minimize the amount of resource that use to match with wrong image. Next, the program uses FlannBasedMatcher provided by OpenCV library to match the input keypoint descriptor with the trained keypoint descriptor. After successfully performed the matching, some calculations had been done to eliminate the false matches. The calculated score which meet the minimum required score and it will bring forward to the next stages.

At the second stage, the program will rematch the selected keypoint descriptor from previous stages. The program uses the original dimension of the image to match again with the trained images. It was to ensure the result is accurate. As mentioned at the first stage, the program uses the FlannBasedMatcher to match the input keypoint descriptor with the trained keypoint descriptor. It also undergoes the verification step to ensure the matching result fulfil the minimum required score. If it is a successful match, the program will continue process the following steps.

f) Image Specification

After successfully matching the image, the program will use findHomography function in OpenCV to find the transformation between matched keypoints. It is used to locate the room in the images. After that, the program will use the perspectiveTransform function from OpenCV library to map the points. The result of transform was a set of four corners which point out the matched area.

There are two types of results from the transformation. If there will be the room specification, the program will return the 4 corners' position of each respective room. For example, a Faculty of General Office may locate on certain part in the building, the system is able to detect which area of it as shown in Figure 3-11. While, if there was no any room specification, the program will indicate which one was the matched building in the images as shown in Figure 3-12. The result will send back to the web system.

Figure 3-11: Office Found in Building Result

Figure 3-12: Building Found Result

g) Building Information Acquisition

After successfully recognizing the landmark, respective building information will be retrieved from the database. The building information which includes the building name, department within the building, department phone number and department website. The information are encoded in JavaScript Object Notation (JSON) type in order to provide a more convenient to read and write.

h) Building Information Responds

The encoded building information will be sent back to client side using the PHP socket. After it is successfully sent out, the client side processing the data and display it by overlapping the data on the real time image in the smartphone.

3-1-3 Database Design

In the building recognition system, it needs a database to manage the training data where the data will be used for recognition process. Hence, a database should be designed to fulfil all the requirements of the system. The data dictionary for this database is constructed and as shown in Table 3-1. Figure 3-13 shows the entity relational diagram (ERD).

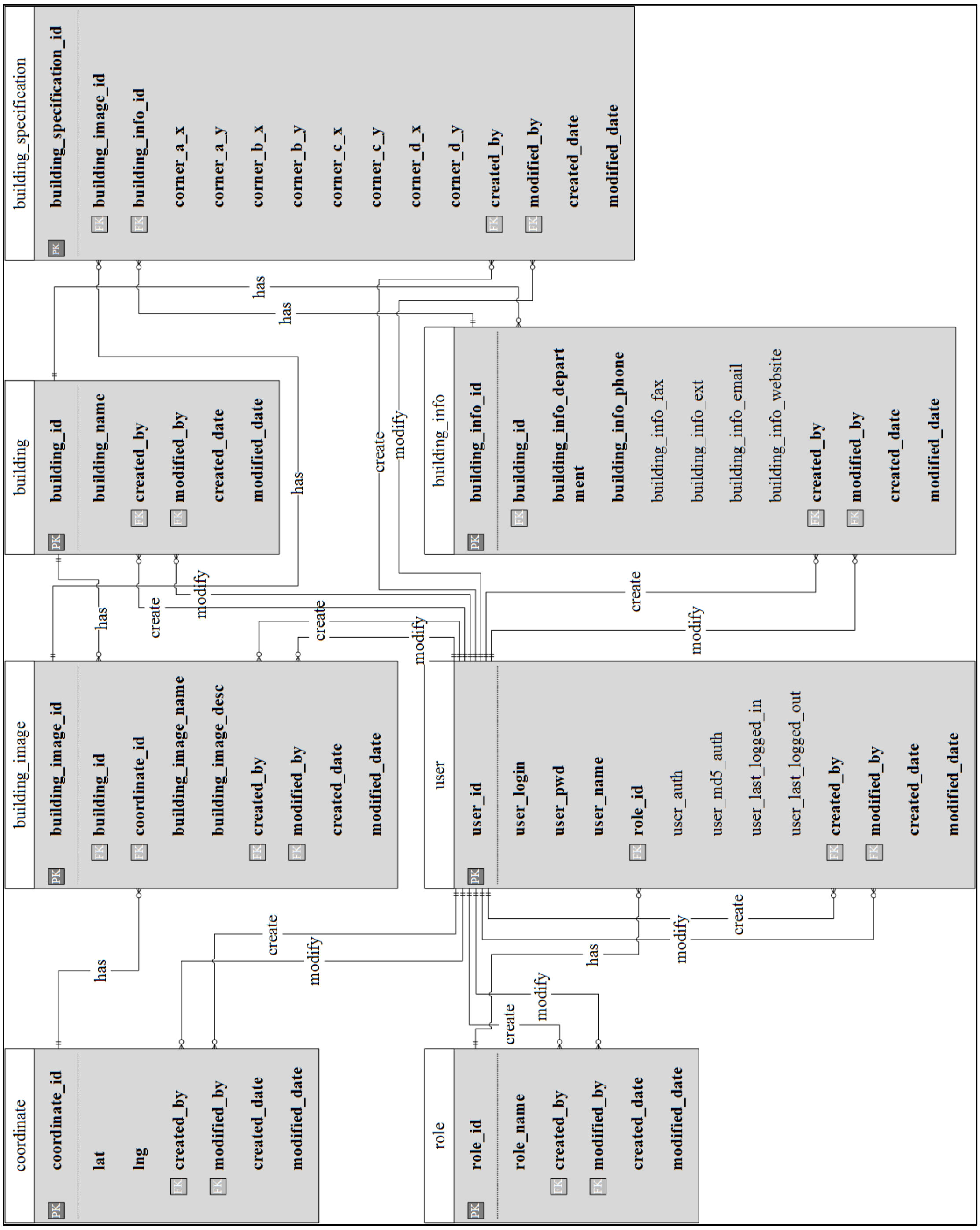


Figure 3-13: Entity Relational Diagram (ERD)

Table 3-1: Data Dictionary

Table Name	Attribute Name	Content	Type	Format	Required	PK@FK
Role	role_id	Role id	Integer(10)	99999	Yes	PK
	role_name	Role name	Varchar(150)	Xxxxx	Yes	
	created_by	Role creator	Integer(10)	99999	Yes	
	modified_by	Role modifier	Integer(10)	99999	Yes	
	created_date	Role created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Role modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
user	user_id	User id	Integer(10)	99999	Yes	PK
	user_login	User login name	Varchar(25)	Xxxxx	Yes	
	user_pwd	User login password	Varchar(45)	Xxxxx	Yes	
	user_name	User name	Varchar(50)	Xxxxx	Yes	
	role_id	User role	Integer(10)	99999	Yes	FK
	user_auth	User authenticate code	Varchar(159)	Xxxxx	No	
	user_md5_auth	User authenticate code (MD5)	Varchar(150)	Xxxxx	No	
	user_last_logged_in	User last logged in date and time	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	user_last_logged_out	User last logged out date and time	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	created_by	User creator	Integer(10)	99999	Yes	
	modified_by	User modifier	Integer(10)	99999	Yes	
	created_date	User created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	User modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
building	building_id	Building id	Integer(10)	99999	Yes	PK
	building_name	Building name	Varchar(500)	Xxxxx	Yes	
	created_by	Building creator	Integer(10)	99999	Yes	
	modified_by	Building modifier	Integer(10)	99999	Yes	
	created_date	Building created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Building modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
building_info	building_info_id	Building information id	Integer(10)	99999	Yes	PK
	building_id	Building id	Integer(10)	99999	Yes	FK
	building_info_department	Building department	Varchar(250)	Xxxxx	Yes	
	building_info_phone	Building department telephone	Varchar(50)	Xxxxx	Yes	
	building_info_fax	Building department fax	Varchar(50)	Xxxxx	Yes	
	building_info_ext	Building department extension	Varchar(50)	Xxxxx	Yes	
	building_info_email	Building department email	Varchar(250)	Xxxxx	Yes	
	building_info_website	Building department website	Varchar(250)	Xxxxx	Yes	
	created_by	Building information creator	Integer(10)	99999	Yes	
	modified_by	Building information modifier	Integer(10)	99999	Yes	
	created_date	Building information created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Building information modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
coordinate	coordinate_id	Building coordinate id	Integer(10)	99999	Yes	PK
	lat	Building latitude value	Double(10,6)	99.999999	Yes	
	lng	Building longitude value	Double(10,6)	99.999999	Yes	
	created_by	Building coordinate creator	Integer(10)	99999	Yes	
	modified_by	Building coordinate modifier	Integer(10)	99999	Yes	
	created_date	Building coordinate created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Building coordinate modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
building_image	building_image_id	Building image id	Integer(10)	99999	Yes	PK
	building_id	Building id	Integer(10)	99999	Yes	FK
	coordinate_id	Building coordinate id	Integer(10)	99999	Yes	FK
	building_image_name	Building image file name	Varchar(150)	Xxxxx	Yes	
	building_image_desc	Building descriptor file name	Varchar(150)	Xxxxx	Yes	
	created_by	Building image creator	Integer(10)	99999	Yes	
	modified_by	Building image modifier	Integer(10)	99999	Yes	
	created_date	Building image created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Building image modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
building_specification	building_specification_id	Building specification id	Integer(10)	99999	Yes	PK
	building_image_id	Building image id	Integer(10)	99999	Yes	FK
	building_info_id	Building information id	Integer(10)	99999	Yes	FK
	corner_a_x	Corner A x-axis	Integer(10)	99999	Yes	
	corner_a_y	Corner A y-axis	Integer(10)	99999	Yes	
	corner_b_x	Corner B x-axis	Integer(10)	99999	Yes	
	corner_b_y	Corner B y-axis	Integer(10)	99999	Yes	
	corner_c_x	Corner C x-axis	Integer(10)	99999	Yes	
	corner_c_y	Corner C y-axis	Integer(10)	99999	Yes	
	corner_d_x	Corner D x-axis	Integer(10)	99999	Yes	
	corner_d_y	Corner D y-axis	Integer(10)	99999	Yes	
	created_by	Building specification creator	Integer(10)	99999	Yes	
	modified_by	Building specification modifier	Integer(10)	99999	Yes	
	created_date	Building specification created date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	
	modified_date	Building specification modified date	Datetime	YYYY-MM-DD HH:MM:SS	Yes	

CHAPTER 4 METHODOLOGY AND TOOLS

4-1 Methodology

A suitable methodology is important for system development project where it can ensure the project complete successfully within a fixed time. Furthermore, developers can develop the system by following the procedure from the methodology that they use. Hence, a prototyping methodology as shown in Figure 4-1 will be used for this system development. The reason to choose prototyping as the methodology for this system is it can reduce time and cost while increases users' involvement into this project. Besides that, it is able to get feedback from users to further enhance the system.

First of all, the development will go through the planning phase which will output a plan how the system will be developed. Next, analysis, design and implementation stages will be gone through and it will produce a system prototype. The system prototype will be reviewed by users and the system will be modified based on users' feedback. The processes will be repeated continuously until it fulfils the users' requirement. Finally a system will be fully developed.

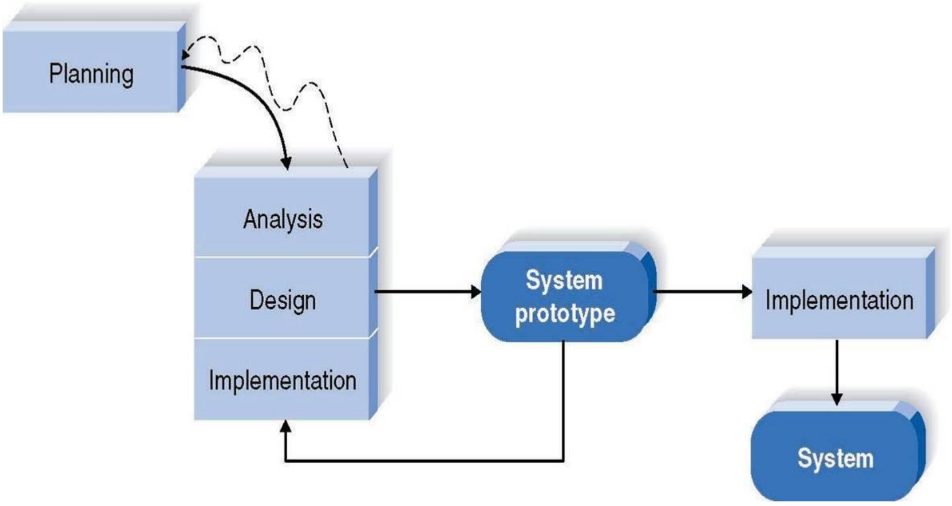


Figure 4-1: System Development Life Cycles

4-2 Tools

4-2-1 Hardware Specification

The hardware specification used throughout the project as shown in Table 4-1.

Table 4-1: Hardware Specification

Processor	Intel® Core™ i5-3230M CPU@2.60GHz
Operating System	Windows 8.1 64bits
Memory (RAM)	8 Gigabyte (GB) DDR3-800
Graphic Card	NVIDIA GeForce GT740M

4-2-2 Software Specification

The software specification used throughout the project stated as below.

a) Microsoft Visual Studio 2013

It is an integrated development environment (IDE) software application and it is used to develop computer software. It simplifies the task of creating, debugging and running the developed application. It is used to develop the recognition system. Since this system need an integrated development environment to develop, hence this software will be used to program during the project period.

b) Netbeans IDE

It is an open source IDE software application and it is used to develop a web applications in this project. It is fast and supports many languages such as PHP. It ease the task for efficient project management where it shows a clear overview of large applications with thousands of folders and files. Hence, it is easy to understand the structure of the applications because all the files and folders are clearly shown. It will be used to develop PHP applications in this project.

c) Open Source Computer Vision Library (OpenCV)

It is an open source library of programming which provides computer vision methods with C++ language. It can be used to perform different types of image processing on the applications.

In this recognition system, it will use the function provided by OpenCV to perform certain tasks. For example, the system will use functions to perform keypoint finding process. It will be more convenient to implement particular task. Hence, the system mostly will use the functions provided by OpenCV to perform any task when needed.

d) MySQL

It is an open source relational database management system (RDBMS) which provides a system to manage a database easily. It will manage the data such as saving the data into the database, retrieving data from database and other database operation.

In this recognition system, it will use MySQL to manage the database. The database contains all the data that is needed for recognition process later. Since there are a lot of data, it need a system to manage the database and retrieve related data easily by filtering the data. It will be easier to execute the recognition process with accurate data.

e) XAMPP

It is an open source PHP development environment which will be used in this project to run the PHP applications that created. It also consists MySQL which can easily manage the database with user friendly interface.

4-3 Requirements

The system is able to recognize the building within certain environment factor. There is only a certain suitable light intensity that the system can recognize. The system unable to recognize the building at night time due to the light intensity is low. This is when the light intensity is low, the captured image is dark and hard to find out the feature points of the building. However, when the light intensity is too high also cannot be accepted in this recognition process. The captured image will be too bright and the system hard to identify the feature points too.

Besides that, the network transmission must be fast to enable the mobile navigation system to communicate with the server to perform their building recognition task. Since the system need high rate of communication between server and mobile, the connection must strong and fast. These 2 devices need internet connection as both need to retrieve coordinates data from GPS to perform recognition process.

Next, the quality of the mobile camera must at least QVGA (320X240) preview resolution to capture a clear image. The system require pixel dimension 640 pixels X 480 pixels to compare all the same dimension image descriptor. This can prevent the system from unable to carry out the recognition process.

Lastly, the processing power of the server must fast enough to process all the request from mobile. Since the real time application need a fast response from the server, the server hardware must have high technology which able process the request in a short time. If the time is too long, the mobile user may not depend the applications anymore.

4-4 Verification Plan

The current system is able to recognize the landmark with high accuracy, high consistency and fast. The quality of the input landmark picture is an important factor that will affect the recognition result. It may be affected by different situation and it may result in false detection and other problem when performing recognition. Few situations are explained as below:

a) The received landmark picture is taken under a low or high light intensity environment condition

b) The received landmark picture is taken in an overlapping scene

c) The received landmark picture is in different scale

d) The received landmark picture is rotated few degree

Therefore, there will be a few verification plan to ensure the accuracy and consistency while recognizing the landmark and stated as follow:

a) Low or high light intensity environment condition

Table 4-2: Verification Plan P1

Procedure Number	P1
Method	Testing
Applicable Requirements	Recognize a low light intensity landmark image
Purpose/Scope	To recognize a low light intensity landmark image

Items Under Test	Landmark image
Precautions	The landmark data must exists in database
Special Conditions/Limitations	If the landmark data does not exist in the database, it will output an inaccurate result
Equipment/Facilities	Laptop
Data Recording	None
Acceptance Criteria	The image is recognized successfully
Procedure	Upload a low light intensity landmark image into server Perform recognition by matching the landmark image with the image data in database
Troubleshooting	Repeat the procedure
Post-Test Activities	None

b) Overlapping scene

Table 4-3: Verification Plan P2

Procedure Number	P2
Method	Testing
Applicable Requirements	Recognize an overlapping scene landmark image
Purpose/Scope	To recognize an overlapping scene landmark image
Items Under Test	Landmark image
Precautions	The landmark data must exists in database
Special Conditions/Limitations	If the landmark data does not exists in the database, it will output an inaccurate result
Equipment/Facilities	Laptop
Data Recording	None
Acceptance Criteria	The image is recognized successfully
Procedure	Upload an overlapping scene landmark image into server Perform recognition by matching the landmark image with the image data in database
Troubleshooting	Repeat the procedure

Post-Test Activities	None

c) Different Scale

Table 4-4: Verification Plan P3

Procedure Number	P3
Method	Testing
Applicable Requirements	Recognize an landmark image with different scale
Purpose/Scope	To recognize an landmark image with different scale
Items Under Test	Landmark image
Precautions	The landmark data must exists in database
Special Conditions/Limitations	If the landmark data does not exists in the database, it will output an inaccurate result
Equipment/Facilities	Laptop
Data Recording	None
Acceptance Criteria	The image is recognized successfully
Procedure	Upload a different scale landmark image into server Perform recognition by matching the landmark image with the image data in database
Troubleshooting	Repeat the procedure
Post-Test Activities	None

d) Rotation

Table 4-5: Verification Plan P4

Procedure Number	P4
Method	Testing
Applicable Requirements	Recognize a rotated landmark image
Purpose/Scope	To recognize a rotated landmark image
Items Under Test	Landmark image
Precautions	The landmark data must exists in database
Special Conditions/Limitations	If the landmark data does not exists in the database, it will output an inaccurate result

Equipment/Facilities	Laptop
Data Recording	None
Acceptance Criteria	The image is recognized successfully
Procedure	Upload a rotated landmark image into server Perform recognition by matching the landmark image with the image data in database
Troubleshooting	Repeat the procedure
Post-Test Activities	None

CHAPTER 5 IMPLEMENTATION AND TESTING

5-1 Overview

In order to test the accuracy and reliable of the building recognition system, several scenarios had been conducted to test the system. They were:

1) Matching in different environment
    a. Light intensity
    b. Scaling
    c. Overlapping
    d. Rotating
2) Transmission between mobile and server
3) Coordinate filtering

The testing process is to ensure the final system is able to provide accurate building recognition. To achieve a higher accuracy of the recognition, the system is able to provide the correct building name and image specification, providing that the environment must sufficient to undergo the recognition process.

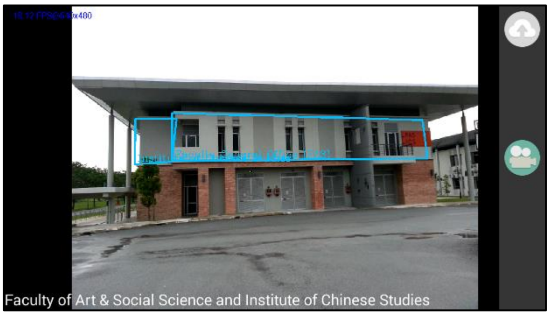
The testing had been conducted at two building which are:

a) Faculty of Art & Social Science and Institute of Chinese Studies, UTAR
b) Faculty of Information & Communication Technology and IPSR Labs, UTAR

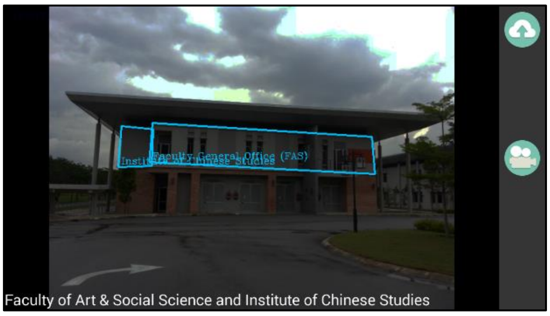
5-2 Matching in Different Environment

The result of testing on different environment for the system is as shown in the Table 5-1.

Table 5-1: Different Environment Testing Results

<table>
<tr><td colspan="2">Normal Environment</td></tr>
<tr><td colspan="2">Faculty of Art & Social Science and Institute of Chinese Studies</td></tr>
<tr><td>
</td><td>Results:

The building is successfully recognized. There are two rectangles indicating that area belongs to an office as shown in the figure.</td></tr>
<tr><td colspan="2">Faculty of Information & Communication Technology and IPSR Labs</td></tr>
<tr><td>
</td><td>Results:

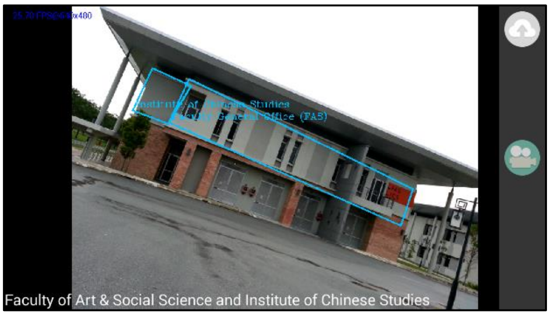
The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.</td></tr>
<tr><td colspan="2">Light Intensity</td></tr>
<tr><td colspan="2">Faculty of Art & Social Science and Institute of Chinese Studies</td></tr>
<tr><td>Low Light Intensity:

</td><td>Results:

The building is successfully recognized. There are two rectangles indicating that area belongs to which office as shown in the figure.</td></tr>
</table>

High Light Intensity 	Results: The building is successfully recognized. There are two rectangles indicating that area belongs to an office as shown in the figure.
Faculty of Information & Communication Technology and IPSR Labs	
Low Light Intensity: 	Results: The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.
High Light Intensity 	Results: The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.

<table>
<tr><th colspan="2">Scaling</th></tr>
<tr><td colspan="2">Faculty of Art & Social Science and Institute of Chinese Studies</td></tr>
<tr><td>Reducing scale:
</td><td>Results:
The building is successfully recognized. There are two rectangles indicating that area belongs to an office as shown in the figure.</td></tr>
<tr><td>Increasing Scale:
</td><td>The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.</td></tr>
<tr><td colspan="2">Faculty of Information & Communication Technology and IPSR Labs</td></tr>
<tr><td>Reducing scale:
</td><td>Results:
The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.</td></tr>
</table>

<table>
<tr><td>Increasing Scale:
</td><td>Results:
The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.</td></tr>
<tr><td colspan="2">Overlapping</td></tr>
<tr><td colspan="2">Faculty of Art & Social Science and Institute of Chinese Studies</td></tr>
<tr><td></td><td>Results:
Not able to recognize the building
Reason:
The tree and the light pole overlaying the building and causes the system unable to recognize it</td></tr>
<tr><td colspan="2">Faculty of Information & Communication Technology and IPSR Labs</td></tr>
<tr><td></td><td>Results:
Not able to recognize the building
Reason:
The tree and the light pole overlaying the building and causes the system unable to recognize it</td></tr>
</table>

<table>
<tr><th colspan="2">Rotating</th></tr>
<tr><td colspan="2">Faculty of Art & Social Science and Institute of Chinese Studies</td></tr>
<tr><td>
</td><td>Results:

The building is successfully recognized. There are two rectangles indicating that area belongs to an office as shown in the figure.</td></tr>
<tr><td colspan="2">Faculty of Information & Communication Technology and IPSR Labs</td></tr>
<tr><td></td><td>Results:

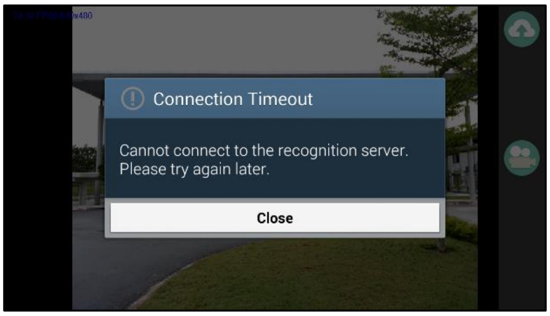
The building is successfully recognized. There is a rectangle to indicate that area belongs to a building as shown in the figure.</td></tr>
</table>

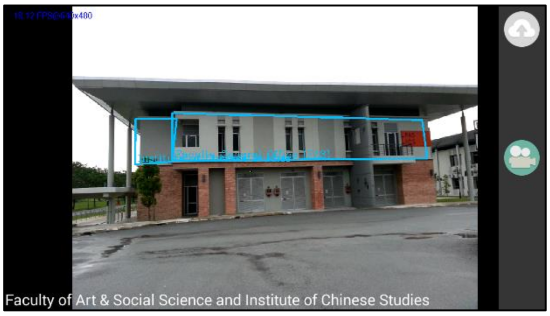
5-3 Transmission between Mobile and Server

There is a connection between mobile and server which requires a network to connect it. For this transmission testing, a router had been setup and it be the middle hardware to transmit all the data between mobile and server. There are few result based on few condition as shown in Table 5-2.

Table 5-2: Transmission between Mobile and Server results

<table>
<tr><th>Cases</th><th>Results</th></tr>
<tr><td>Router is off

Result: The mobile cannot connect to the server and will display connection timeout.</td><td>
</td></tr>
</table>

Router is on Result: The mobile successful connected to server and perform building recognition.	

5-4 Coordinate Filtering

The building navigation system had provided coordinate filtering which will filter all the request based on the coordinate. The ideal circle's radius is 50 meters which means that the system will recognize with those data within the 50 meters. The shaded area as shown in the Figure 5-1 is stated that the coverage area of the coordinate.

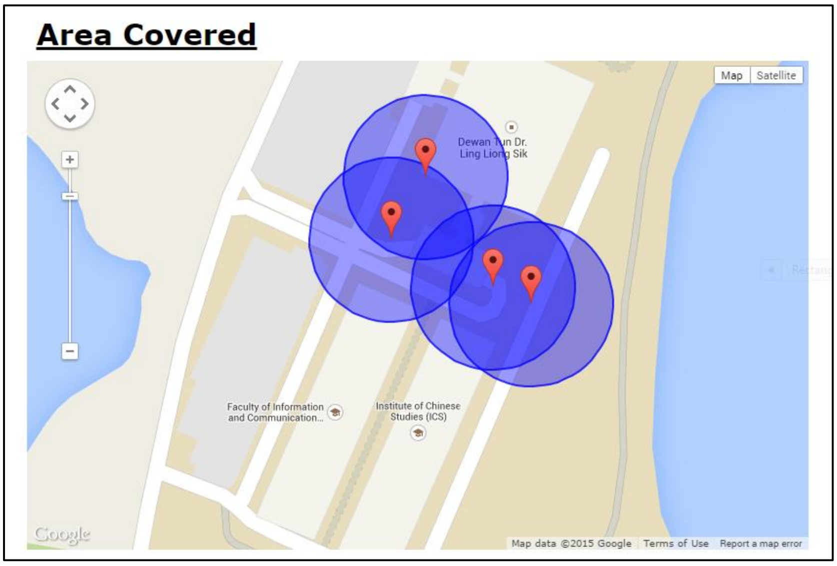


Figure 5-1: Shaded Area Results

A testing had been conducted by collaborating with mobile navigation system. First of all, the mobile user tests it at a coordinate which does not exist in the system. The result of it is successfully when comparing the coordinate with the data in database and reply a message as shown in Figure 5-2.

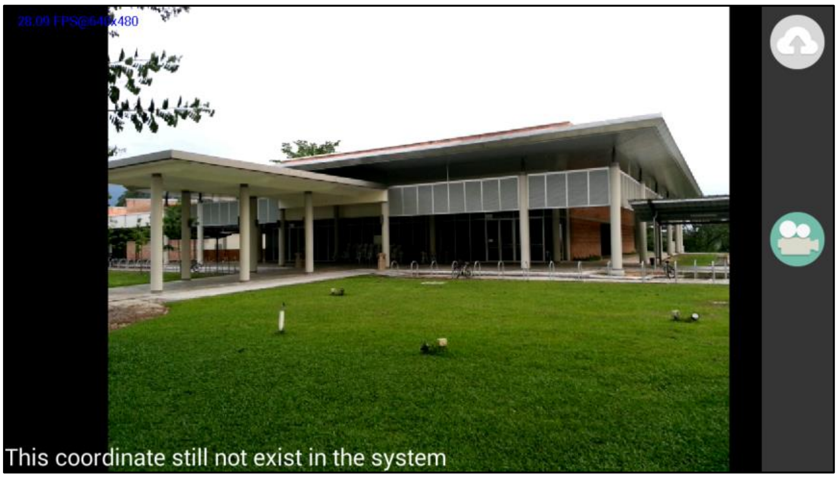


Figure 5-2: Coordinate not exist result

Secondly, the mobile user tests it at a coordinate which are exist in the system. The building recognition system is able to compare the coordinate in database and start to perform recognition on the image as shown in Figure 5-3.

Figure 5-3: Coordinate exist result

CHAPTER 6 CONCLUSION

6-1 Conclusion

In conclusion, this project had develop a building recognition system which is able to recognize an input landmark image that is obtained from the client side and provide the landmark information to the mobile navigation system accurately. Since the system must ensure its accuracy and efficiency, a few techniques had been applied in order to solve the problem. First, SIFT is selected as the recognition algorithm due to its good stability and high efficiency. It is able to recognize an object without affected by the image quality or the degree of view. Therefore, it is used as a recognition algorithm in the building recognition system.

Furthermore, in order to provide a real time response building recognition system, the system will undergo data filtering on the trained keypoint descriptor by using coordinate to ensure fast response time. Besides that, it also uses a descriptor image to perform matching where it can save the time to extract the keypoint again from the trained image. The system also uses MySQL as the database management system where it can manage the trained building data and ease the task of retrieval with faster speed and higher accuracy. Therefore, the system can use all the benefits mentioned above to achieve real time response.

Last but not least, the building recognition system is connected with the mobile navigation system in order to provide a navigation to the users inside the campus area. The UTAR SmartGuide with the combination of building recognition system and mobile navigation system can help the users to understand the building easily and accurate.

6-2 Limitation

In this project, there are several problems exist that may affect the building recognition result. One of the problems is the environment factor when taking the real time image. When the light intensity of the environment is too low or too high, it will make the keypoint descriptor finding produces an inaccurate result. Besides that, the quality of image will cause some issue in this project. Since the input images from client side are taken by phone camera and some phone camera resolution is low, the quality of the image may be affected such as the existence of image noises or low pixels size. The noises will influence the image descriptor by producing incorrect descriptor and cause some inaccuracy in the results.

Furthermore, the server respond time is another problem in this project. Since this project need to run some C++ program inside the server through PHP website, the execution time of the program may be affected by the server processing power and the speed of the internet. Thus, it will affect the time to respond back to client side and the efficiency of the navigation system will be degraded. So, a server with high computational power and high speed internet is a need.

6-3 Future Works

In order to provide a more user friendly and practicable application, some functions should be improved in more convenient ways as explained below:

a) Processing Power

Since the program needs a high performance server to execute and the problem is that when high amount of users use the system at the same time. The server may need a lot of resource to process the request but the resource is limited due to the hardware and software problem.

b) High network transmission speed

The mobile and server need a high performance in network transmission. Both of them need to request and response in real time. Therefore, the system may setup in the server which with a high speed internet connected.

c) Image filtering algorithms

The system can input different types of building image, but it may consume a lot of computer resources when the system performing recognition on all the same images. The images should undergo a filtering step before it input into the server database.

CHAPTER 7 REFERENCES

Bay, H., Ess, A., Tuytelaars, T. & Gool, L. V., 2008. Speeded-up robust features (surf). *Computer Vision and Image Understanding,* 110(3), pp. 346-359.

Gammeter, S. et al., 2010. *Server-side object recognition and client-side object tracking for mobile augmented reality.* San Francisco CA, IEEE.

Li, J. & Allinson, N., 2013. Building recognition using local oriented features. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS,* 9(3), pp. 1697-1704.

Lowe, D. G., 2004. *Distinctive Image Features from Scale-Invariant Keypoints.* Canada, International Journal of Computer Vision.

Rublee, E., Rabaud, V., Konolige, K. & Bradski, G., 2011. *ORB: an efficient alternative to sift or surf.* Barcelona, IEEE.

Tao, Y. et al., 2010. *Performance evaluation of sift-based descriptors for object recognition.* Hong Kong, International MultiConference of Engineers and Computer Scientists.

Zhang, W. & Kosecka, J., 2005. *Localization based on building recognition.* Washington DC, IEEE.