DEVELOPMENT OF SALIENCY METRIC FOR AUTONOMOUS LANDMARK SELECTION IN COGNITIVE ROBOT NAVIGATION

GAO HANYANG

FACULTY OF COMPUTER SCIENCE AND
INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

DEVELOPMENT OF SALIENCY METRIC FOR AUTONOMOUS LANDMARK SELECTION IN COGNITIVE ROBOT NAVIGATION

GAO HANYANG

DISSERTATION SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

UNIVERSITI MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of the candidate: **Gao Hanyang**

Registration/Matric No: **WOA180013 (17198526/1)**

Name of the Degree: **Master of Computer Science (Applied Computing)**

Title of Dissertation: **Development of Saliency Metric for Autonomous Landmark Selection in Cognitive Robot Navigation**

Field of Study: **Image Processing**

I do solemnly and sincerely declare that:

(1) I am the sole author /writer of this work;

(2) This work is original;

(3) Any use of any work in which copyright exists was done by way of fair dealings and any expert or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship has been acknowledged in this Work;

(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;

(5) I hereby assign all and every rights in the copyrights to this Work to the Universiti Malaya (UM), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained actual knowledge;

(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether internationally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature: Date: 25.01.2022

Subscribed and solemnly declared before,

Witness Signature: Date: 25.01.2022
Name:
Designation:

DEVELOPMENT OF SALIENCY METRIC FOR AUTONOMOUS LANDMARK SELECTION IN COGNITIVE ROBOT NAVIGATION

ABSTRACT

Urban landmarks are spatial features that are visually significant in the neighbourhood. Humans cognitively select landmarks based on their visual appearance like size, colour, and shape. Many researchers have attempted to evaluate and extract visual landmarks, usually by abstracting their features and quantifying their salience. Humans use qualitative, high level visual features of landmarks in their navigation. In contrast, robots use empirical, low-level HOG and SURF features in their landmark extraction for navigation. A quantitative model for visual salience indicators in urban landmark extraction seems beneficial to the robotics community and could improve understanding for cognitive robot navigation. Quantifying visual salience indicators for urban landmark extraction is challenging when the goal is to compute qualitative, high-level visual features. Existing robot landmark extraction methods are based on low-level features like HOG and SURF, which fails to express landmarks cognitively like humans. This dissertation proposes an algorithm to quantify urban landmarks based on visual salience indicators for cognitive robot navigation. The dissertation follows three objectives; to segment urban landmarks in an image, to develop an algorithm to quantify visual salience indicators for urban landmarks extraction, and to compare the performance of proposed algorithm in extracting urban landmarks between robot and human. A drone is used to collect fourteen aerial images of urban landmarks. Four images are taken from top view and for variation, one image is taken from front view, for each landmark. The images processing follows bilateral filtering, Otsu thresholding, morphing to resolve connectedness issues, and segmenting the landmarks. Next, the size, colour and shape salience equations are considered following pixel counting, extracting intensity value from hue, saturation and value (HSV), and an equation for shape indicator, respectively.

The experiment done suggests that the final salience value for each landmark can be calculated by adding size, colour and shape together according to weightage 45%, 35% and 20% respectively. Sixty participants between the age of 18 and 60 agree to answer a survey in evaluating 14 urban landmarks based on their size, colour and shape. Encouragingly, 12 out of the 14 urban landmarks selected by the robot match the human selection, with 85.7% accuracy.

Keywords: Visual saliency metric, urban landmark, automated landmark selection, cognitive robotics, image processing

PEMBANGUNAN METRIK UNTUK PEMILIHAN BERAUTONOMI MERCU TANDA TERLIHAT DALAM NAVIGASI ROBOT KOGNITIF

ABSTRAK

Mercu tanda merupakan ciri ruang yang terlihat secara visual di kawasan kejiranan bandar. Manusia secara kognitif memilih mercu tanda berdasarkan penampilan visual mereka seperti ukuran, warna, dan bentuk. Ramai penyelidik telah berusaha untuk menilai dan mengekstrak mercu tanda visual, biasanya dengan mengabstrak ciri-cirinya dan membangunkan algoritma keterlihatannya. Manusia bergantung kepada visual kualitatif mercu tanda, ciri tahap tinggi, dalam navigasi mereka. Sebaliknya, robot menggunakan ciri empirikal HOG dan SURF, tahap rendah, dalam pengekstrakan mercu tanda untuk navigasi. Model kuantitatif bagi indikator keterlihatan visual mercu tanda bermanfaat bagi komuniti robotik meningkatkan pemahaman navigasi robot kognitif. Penilaian indikator keterlihatan visual untuk pengekstrakan mercu tanda bandar adalah sukar apabila tujuannya adalah untuk mengira ciri visual tahap tinggi yang kualitatif. Kaedah pengekstrakan mercu tanda robot pada ciri tahap rendah seperti HOG dan SURF gagal mengekspresikan mercu tanda secara kognitif seperti manusia. Disertasi ini mencadangkan algoritma untuk robot kognitif menilai mercu tanda bandar berdasarkan persepsi manusia untuk indikator keterlihatan visual. Disertasi ini mempunyai tiga objektif; segmentasi mercu tanda bandar dalam imej, pembangunan algoritma yang menilai indikator keterlihatan visual untuk pengekstrakan mercu tanda bandar, dan perbandingan prestasi algoritma antara robot dan manusia dalam pengestrakan mercu tanda bandar. Sebuah dron digunakan untuk mengumpulkan empat belas imej udara mercu tanda bandar. Untuk setiap mercu tanda, empat imej diambil dari pandangan atas dan untuk variasi, satu imej diambil dari pandangan depan. Pemprosesan imej mengikuti langkah-langkah seperti penapisan dua hala, ambang Otsu, morfing untuk menyelesaikan masalah keterhubungan, dan segmentasi mercu tanda. Seterusnya, persamaan ukuran,

warna dan bentuk dipertimbangkan berdasarkan penghitungan piksel, mengekstrak nilai intensiti dari rona, tepu dan nilai (HSV), dan persamaan untuk penunjuk bentuk. Eksperimen yang dilakukan menunjukkan bahawa nilai keterlihatan setiap mercu tanda dapat dikira dengan menambahkan ukuran, warna dan bentuk bersamaan mengikut pemberatan 45%, 35% dan 20%. Enam puluh peserta berusia antara 18 dan 60 tahun bersetuju untuk menjawab tinjauan dalam menilai 14 mercu tanda bandar berdasarkan ukuran, warna dan bentuknya. Keputusan yang menggalakkan dimana 12 dari 14 mercu tanda bandar yang dipilih oleh robot bersamaan dengan pilihan manusia, dengan ketepatan 85.7%.

Kata kunci: Metrik keterlihatan visual, mercu tanda bandar, pemilihan mercu tanda automatik, robotik kognitif, pemprosesan imej

ACKNOWLEDGEMENTS

In successful completion of this work, I would like to express my deepest appreciations to my kind supervisor Dr Zati Hakim Azizul Hasan for her insightful suggestions and her valuable time. Without her selfless help, I could not successfully finish the dissertation during this tough period. She helped me a lot not only in my study but also in my life. During this hard time (corona virus), she taught me how to write a better dissertation and how to make better presentations as well as how to use useful tools to do research. I also want to express my deepest respect to Prof Dr Loo Chu Kiong and Dr. Muhammad Shahreeza Safiruz Kassim for their constructive advice and encouragement. No words could express my admiration and sincere gratitude to my supervisor and panels, who encouraged and inspired me when I encountered difficulties. Their suggestions were always valuable, and their technical comments lead to the completion of this research project.

Finally, my deepest thanks are expressed to all my family members for their encouragement and support, especially my kind parents and girlfriend, Yiting.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Background

Navigation is a basic but one of the most critical parts for animals. It helps them from an initial location to a goal. Without making wise decisions on routes taken from original places to destinations, animals may have trouble surviving. Navigation activities, to human beings, always involve planning which means determining a destination, orientation, and reorientation during travel when encountered obstacles. Humans and animals rely on a kind of mathematical model of their belief systems, representing a draft copy of one's surrounding information, called a "cognitive map" (Axelrod, 2015), to perform navigation activities.

A cognitive map is essential for an animal and human to traverse exclusively indoors and outdoors. Such a map enables one to take the optimal route from a given location to a known destination, and environmental perceptions naturally influence this map. A study shows that the factors are related to the navigation abilities of humans and animals (Wolbers & Hegarty, 2010). According to the research, the human and animal model of cognitive mapping results from analysing specific spatial information from the external environment and translating them as an internal representation. What this internal representation looks like in mind is the focus of many researchers, including behavioural scientists and roboticists. But one feature remains dominant in both mapping discussions: the cognitive map contains landmarks (Epstein et al., 2017; Krupic et al., 2018; Wang et al., 2019; Gupta et al., 2017; Zhao et al., 2019; Dubey et al., 2019).

A landmark is an object that is prominent or, in other words, cognitively attractive to people compared with its surroundings (Weng et al., 2017). In the real world, buildings are usually considered ones' navigation landmarks within urban areas. Those selected urban landmarks are usually unique in shape, built at an attractive spatial location or has historical stories or profound meanings behind them. Landmarks play a significant role in human navigation activities. Landmarks are the basics of cognitive maps and are often used in wayfinding and representation of routes knowledge (Duckham et al., 2010). Selecting an appropriate landmark during navigation remains a hot topic for researchers studying cognitive maps and robot navigation.

Azizul & Yeap (2015) investigate how the cognitive map is built from landmarks by experimenting with a laser sensor using an indoor mobile robot. The basis of their experiment is the notion that landmark is the basis of cognitive map building for humans and animals. A landmark in their work is defined as large 2D surfaces which usually makes up the wall and larger furniture in the indoor environment. In their experiment, they learned that if their laser robot can make an association between the spatial relationship of one landmark to another, they can approximate the rough location of those landmarks to each other and form a map of the surrounding.

Figure 1(a) shows the blueprint of an office environment, and depicted in yellow is the path and direction where the laser robot traversed. After making a round trip through long corridors, the laser robot built a map shown in Figure 1(b) with green depicting the path the robot traversed. By associating the surfaces (or landmarks), the laser robot can build a rough representation of the map resembling, most importantly, the overall shape of the real environment.

Figure 1.1: (a) Blueprint of indoor office environment used for testing in Azizul & Yeap (2015) and (b) the cognitive map built by the laser robot after traversing the office.

The extraction of a 2D landmark such as the ones in Azizul & Yeap (2015) can be helpful to test theories in cognitive mapping. The association of a landmark can lead to developing one's internal representation of the external environment. However, looking at the map generated, one can note that the surfaces surrounding the laser robot look similar. With the laser robot traversing corridor segments with turns, the robot cannot identify one single landmark from one segment and associate it with correcting the position of all the other surfaces in other segments.

So how did the laser robot identify which landmark to associate with at a particular instance? In Azizul & Yeap (2015)'s work, surfaces in each path segment have been upgraded to landmark status, so surfaces in that segment can be spatially associated with one another. For other segments, a new landmark is selected to associate the surfaces. The observation poses an interesting question. How does an object in the surrounding becomes a landmark? What are the criteria for selecting landmarks? Is the status upgrade from object to landmark a straightforward process? Is the case the same for urban landmarks?

Weng et al. (2017) researched selecting urban landmarks based on spatial information. In their work, spatial information is considered to be the dominant factor. A spatial salient urban landmark is at the position that separates urban regional structure, dividing different districts or has high accessibilities to other places. However, they also remark that other salient features like visual appearance (i.e., colour, shape, and size) are strong indicators, which they have not considered. Therefore, selecting landmarks by calculating visual salience features remains an exciting question to solve.

1.2 Motivation

In more recent robotics works, the rise and commodity of visual sensing promote visual information as the primary tool for landmark extraction. Puthussery et al. (2017) used objects detected from the images taken from its cameras as navigation markers. The machine learning method is used to detect and trace the markers. Once the object is observed, it is classified and automatically selected as landmarks. Their work contrasts the fundamentals proposed by Götze & Boye (2016) concerning visual landmarks. In their research, Götze & Boye (2016) suggested that visual landmark selection is a process filtered by visual salience determined by the object's colour, size, and shape. In other words, any object that is not visually distinct compared to other objects in the surrounding should not be upgraded to landmark status and not be considered in the landmark selection.

Other vision-based researchers have begun to investigate algorithms for visual landmark selection (Li et al., 2017; Kunii et al., 2017; Ishikoori et al., 2017). Considering the nature of the visual sensing platform and the available vision algorithms, they proposed different feature gradients like HOG, SURF or used a convolutional neural network to describe a visual landmark.

Furthermore, in Li et al. (2017), the robot is pre-programmed to search for specific indoor objects for a landmark, like the fire extinguishers or doors. It is apparent that since landmarks must be significant and should not be random, most of these works prefer to feed the machine with a priori rather than allowing the machine to deduce a landmark on its own.

In this work, I am interested in investigating Gotze & Boye (2016)'s proposition of visual landmark salience rather than considering spatial and semantic features. Instead of focusing on low-level visual features such as the HOG and SURF, I am interested in evaluating objects in the robot's surroundings through their salient cognitive potential. Is the object the brightest among other objects in the surrounding? Is the object the largest? And, is the object having the most exciting shape among the rest? Which of these features are most or least important? How about when all three salient features are combined? Will the combination make a particular object stand out visually among the rest? This way, I can avoid pre-programming my robot with a fixed landmark because, learning from the human and animal model of cognitive mapping, landmark selection should be an automated process (not a priori) and certainly not by random selection.

To emphasize my solution, allow me to use buildings in the human visual surrounding as examples. To be selected by humans as a landmark, a building usually must be easily recognized (Duckham et al., 2010; Lerner et al., 2007). This means, landmarks usually contain unique information or distinct features compared to their neighbours. For example, in Kuala Lumpur, two of the most famous landmarks are the Twin Towers and Istana Negara (see Figure 1.2). There are the likes of the Forbidden City in China, the White House in the USA and the Leaning Tower of Pisa in Italy. All of them are

particularly unique from other regular and typical buildings. The Twin Towers are unique in terms of height (Barr et al., 2015), and as for the Istana Negara, the bright colour with majestic shape is quite impressive at first sight (Sherif, 2012).

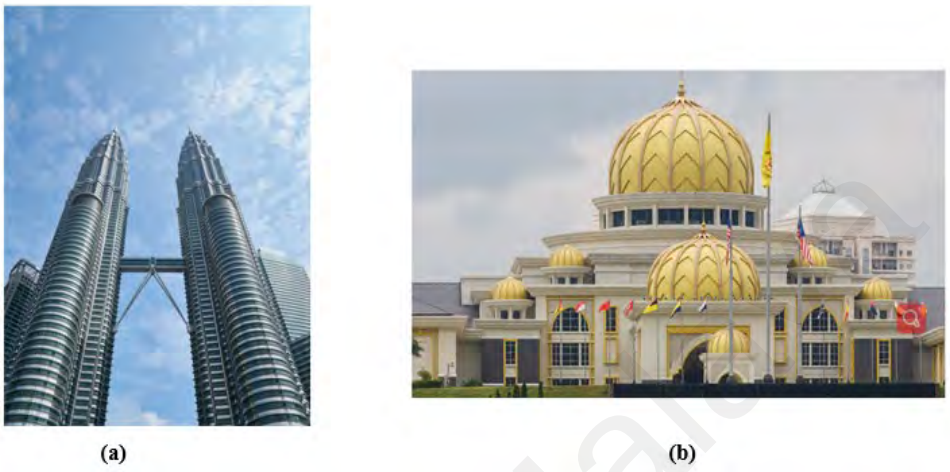


Figure 1.2: Landmarks in KL (a) the Twin Towers (b) Istana Negara

Those buildings show that these visual salience features, colour or brightness, size and shape, collectively, can determine if an object in the surrounding is visually distinct and suitable as a landmark. The same notion can be investigated for robot navigation settings. The difference is, where humans rely on qualitative assessment, a robot requires a metric to measure salience. It is without question that the first two salient visual features, the brightness and size, are straightforward in computation, given the various tools and libraries equipped in image processing.

But calculating whether the shape of an object is attractive from the perspective of humans and robots remains challenging. The likes of Florian et al. (2012) version of empirical evidence for landmark salience needs to be investigated in developing an algorithm for computing landmark salience for autonomous robot navigation. Quantifying visual salience indicators (size, colour and shape) for urban landmarks is challenging when computing those qualitative visual features.

1.3 Problem Statement

Quantifying visual salience indicators for urban landmark extraction is challenging when the goal is to compute qualitative, high-level visual features. Existing robot landmark extraction methods are based on low-level features like HOG and SURF, which fails to express landmarks cognitively like a human. In this work, I proposed an algorithm to quantify urban landmarks based on visual salience indicators, i.e. colour, size and shape, for cognitive robot navigation.

1.4 Research Questions

The following questions have been designed to investigate the problem statement:

1. What are the discriminators that distinct a good landmark for robot navigation?
2. How can salient visual features be combined into a metric? Which feature is more important than the others?
3. What kind of approach is suitable for extracting visual features from the robot environment? Machine learning or hard coding?
4. How does the cognitive robot algorithm perform in urban landmark selection compare to a human?

1.5 Objectives of the Study

To answer the research questions, the following objectives are determined for the study:

1. To segment urban landmarks from images
2. To develop an algorithm to quantify visual salience indicators for urban landmarks extraction
3. To compare the performance of the proposed algorithm to human

1.6 Research Mapping

The research question, objective, method and outcome mapping is as follows:

Table 1.1: RQ, RO, Methodology and Outcome

Research Questions	Research Objectives	Methodology	Research Outcome
What are the discriminators that distinct a good landmark for robot navigation?	To study landmark features and their usefulness in robot navigation	a) Literature search b) Systematic literature review	a) Core criteria for good landmark selection b) The mathematical formula to quantize individual criteria into measurable data
What kind of approach is suitable for extracting visual features? Machine learning or hard coding?	To segment urban landmarks from images	a) Algorithm development	An algorithm for extracting visual landmark features
How can salient visual features be combined into a metric? Which feature is more important than the others?	To develop an algorithm to quantify visual salience indicators for urban landmarks extraction	Model development System development	a) A metric model for visual landmark salience b) A system for autonomous landmark selection
How to improve the consistency of landmark recognition in robot navigation?	To compare the performance of the proposed algorithm in extracting urban landmarks between robot and human	Experiment Validating and fine-tuning	Results of performance on the consistency of landmark recognition

1.7 Scope of the Study

The following scopes have been identified in addressing the research objectives:

1. Require distinct gaps between objects (can be bird's eye view or front view)
2. High objects visibility, no fogs, no obstacle/overlap with other objects
3. The foreground image should have a higher intensity value than the background. And the difference between the average intensity value between foreground and background should be big enough.
4. The objects on the image should have closed shapes.

1.8 Significance of the Study

Developing an algorithm to compute visual landmark salience requires an in-depth study into the individual descriptors that distinguish a good landmark. A metrical model combining these individual descriptors is beneficial to the robotics navigation community, particularly in filling in the gap for a method for autonomous landmark selection. It is hoped that a quantitative solution for distinguishing recognizable landmarks from the environment can improve our understanding of the role of the visual landmark in building cognitive maps.

1.9 Summary

This chapter introduces the dissertation, highlighting the background and critical problems in robot urban landmark extraction. The advancements of image processing motivate the solution proposed. The problem statement and mapping of research questions to objectives, methods, and outcomes clarify this dissertation's direction. The study aims to improve urban landmark extraction for robots by developing a metric based on the visual appearance of urban landmarks. The following chapter describes the literature review done.

CHAPTER 2: LITERATURE REVIEW

2.1 Overview

This chapter describes the review done to address the research questions. The review begins by learning the basics of robot navigation. A significant review is proposed for landmarks and image processing techniques. They get special sections, respectively.

2.2 Robot navigation systems

Navigation is the act of moving from one place to another. Navigation is defined as a procedure to determine the safe and suitable route between an initial and an endpoint for the traveller, be it human or robot. The problem in robot navigation can be defined as two parts: localization and motion. Localization, in general, is the understanding of where the system is. It answers the question, “Where am I”. While motion, which determines the following position based on the current situation, answers the question “Where to go next”. The methods to do localization can be categorized as an accumulative approach. It often combines odometry (Chow et al., 2019), beacon-based (Liskovec & Kovarova, 2016), landmark-based (Loevsky & Shimshoni, 2010) and GPS (Cui et al., 2015).

Several robot control architectures can monitor robots’ behaviour under different circumstances. Those architectures are mainly categorized into four classes: planner-based, purely reactive, hybrid-based and behaviour-based (Isa et al., 2016). A central reasoning unit is embedded for the planner-based robot to process the data collected from its sensors and control its movement. In comparison, a purely reactive robot works more biologically by connecting its perception sensors and actuator actions in animals’ world called classical conditioning.

For example, when a robot realizes a left obstacle, it will immediately turn the right-side wheel forward without waiting for the central control unit to process and send commands instead of planner-based architecture. The notion for the hybrid system is to combine the previous two systems. Reaction control is used for the lower level, while the decision marker is the central processing unit for the higher level. Finally, the behaviour-based approach is quite similar to a purely react system but more complex. The difference is that other conditions may influence the action of this system. Internal states can be considered, and other behaviours can change the react behaviour.

2.3 Urban landmarks and navigation

Landmarks are those objects or buildings with distinct features that can be recognized at first sight. Such landmarks play a vital role in landmark-based navigation systems (Edgar et al., 2012). The efficiency of landmark selection directly influences the efficiency and accuracy of robot navigation. Table 2.1 shows robot navigation systems that are not landmark-based for reference. A landmark-based system is preferred mainly based on two reasons. A cheaper hardware option to embed a camera on the robot and vision offers fast feature recognition towards identifying objects as landmarks.

Table 2.1: Non-landmark systems for robot navigation

	Mechanism	**Economy cost**	**Time cost**
Odometry	Measuring the wheel rotation	Low	Low
Inertial	Using a gyro to indicate each turn	High	Low
Beacon-based	Triangulation, usually three beacons	High	High
GPS	Trilateration required at least three satellites	High	High
Landmark-based	Recognition of desired objects at a known place	Low	Low

2.3.1 Famous urban landmarks around the world

There are some famous landmarks around the world. They are unique in appearance (structure, colour, size) and have a rich cultural history. For this research, objects with distinctive features are paid attention to. And these landmarks are separate apart from their geographical neighbours due to their appearance (see Figure 2.1: World famous landmarks (a) the Eiffel Tower (b) Leaning Tower of Pisa (c) Statue of Liberty (d) Cristo Redentor).

Figure 2.1: World famous landmarks (a) the Eiffel Tower (b) Leaning Tower of Pisa (c) Statue of Liberty (d) Cristo Redentor

2.3.2 Role of landmarks in navigation

Landmarks have been used to describe different contexts using visual information (Edgar et al., 2012). And recent researches on biological spatial navigation study reveal that landmarks can potentially affect animals' navigation behaviour by generating visual stimulus (Tommasi et al., 2012). An interesting experiment shows the relationship between landmarks and rodent navigation (Rodriguez et al., 2014). Figure 2.2 shows an experiment observing rats behaviour and decision making while navigating their environment.

Figure 2.2: The Morris Water Maze paradigm (Rodriguez et al., 2014)

The rats in the experiment are separated into four groups; three groups have a platform visible to the rats, while the fourth platform is submerged by opaque water. The group with hidden platform rats has to use colourful cues, which are landmarks, in this case, to localize the position of the hidden platform. And for each trial, those rats are placed in a different origin place, making egocentric strategies impossible. The result shows that the rats can still find the platform even when the destination is invisible, long as the appropriate landmarks are given, and the relationships between cues are learned.

The previous research explains how landmarks work for rats, but how about robots? What do landmarks mean to robots? According to Ahmadpoor & Shahab (2019), spatial knowledge has different forms: landmark, route and survey knowledge. Landmark knowledge represents several selected objects at a fixed position, while route knowledge is generally line segments connecting landmarks from origin to destination. Survey knowledge represents a set of route knowledge that indicate the potential routes from a place to another. Route landmarks are reproduced from landmark knowledge and spatial information. Figure 2.3 shows a representation of the different spatial information.

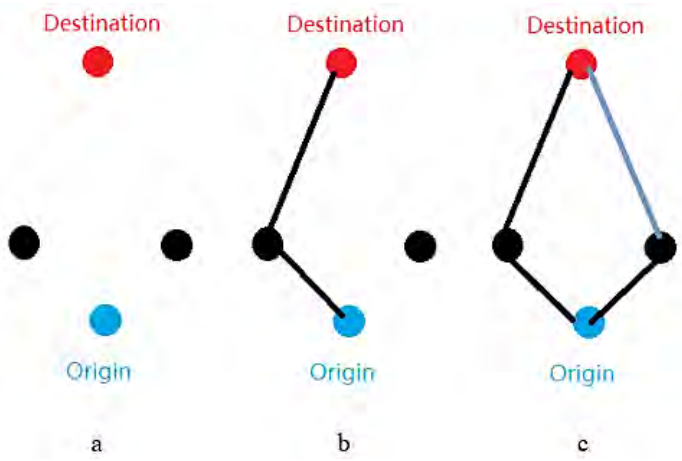


Figure 2.3: Different forms of spatial information: (a) landmark knowledge (b) route knowledge (c) survey knowledge

2.3.3 Landmark saliency modelling

The order in which landmark appears in a navigation process affect the efficiency and accuracy of the navigation. The computation concerning landmarks usually appear three times in a navigation process. The beginning, when a human or robot picks up several landmark candidates. Then immediately filtering the candidates and deciding on one that matches specific criteria—finally, traversing to the landmark. From a navigator's perspective, traversing towards the landmark include recognizing the selected landmark over consecutive images. Using an appropriate detection and recognition algorithm makes this part of the navigation process more precise and timesaving.

2.3.3.1 Landmark selection and salience information

Landmark selection should be based on a reasonable standard like colours and size etc. Some researchers (Edgar et al., 2012; Florian et al., 2012; Clemens et al., 2004) discussed various taxonomy on landmarks identification and the selection of salient features. Locations or regions of interest that are paid more attention to are considered salient, while others are interpreted as background. Visual information like colours, size, shape, scale, and location are considered in calculating a potential landmark's uniqueness. However, developing a salient model for landmarks are not straightforward. According

to Gangaputra (2017), researchers commonly propose salient landmark models based on visual appearance and semantic attraction, while structural attraction is less considered.

The many features that make a landmark visually and structurally attractive are programmable from a machine's perspective. Hence the popularity of the visual and structural based models in configuring landmark saliency. The semantic features are subjective and less easy to program. The visual appearance features evaluating the façade or surface area of a landmark. People are likely to notice the size and shape of the building, so facade area is a fundamental component when measuring salience. It is also hypothetically straightforward to size; a building with a cuboid shape, for example, is a product of width multiple by the height. An image of the building is made up of pixels. By counting the pixels of the desired area, one can get the facade area size.

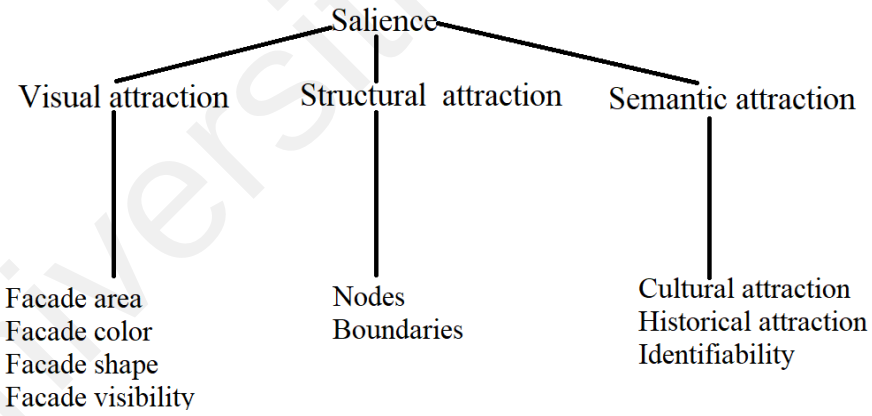


Figure 2.4: Possible landmark salience models (Gangaputra, 2017)

Figure 2.5: Example of an outdoor scene

Besides the facade, colours also play an essential role in salience information. A building with a colour that is different from its surroundings will likely draw immediate attention. A standard method calculates the average value of an image and compares it with the RGB (Red, Green and Blue) value extracted from the candidate object. If the difference exceeds a pre-defining threshold, then it is selected to be the landmark. However, in the physical world, illumination changes over time, drastically impacting computer vision processing. Figure 2.5 is an example of an outdoor situation.

Notice the cloud shadows on the mountains. To computer vision, the shadow and light areas have contrasting RGB values. So relying only on RGB values will give different results in identifying the mountains in this case. The proposed solution is to use a more illumination-tolerant colour model HSV (hue, saturation and value) to eliminate as much as possible the mistakes caused by the light factor (Alshammari et al., 2018).

The method to calculate the salience value of facade shape is more complex. The method used includes observing the shape deviation. The shape deviation is produced from the orthophotograph, representing an orthorectified aerial image that the scale is uniformed. Figure 2.6 shows an example. Finally, the measurement for visibility. The visibility measurement varies from the moving entities. For example, the math for a potential landmark to a pedestrian is calculated using street space (Gangaputra, 2017). Figure 2.7 shows an example of a street space landmark. In comparison, the visibility value may be calculated using corridor space for an indoor mobile robot.

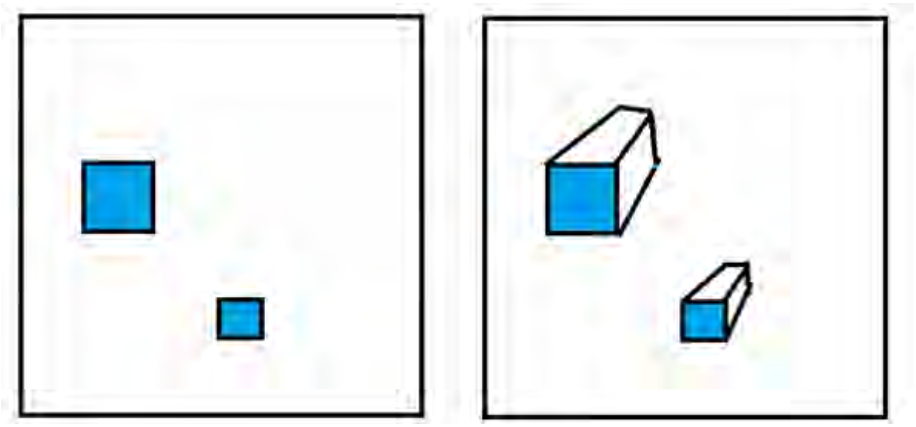


Figure 2.6: Example of orthographical view (left) and the perspective view (right)

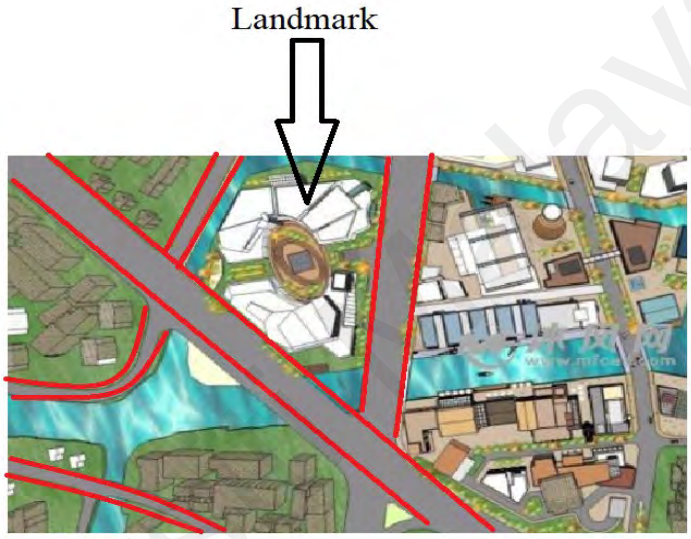


Figure 2.7: Red lines showing visibility calculation (Gangaputra, 2017)

2.3.3.2 Learning landmark saliency from users' route instructions

To derive a mathematical model from pedestrians' route descriptions, Götze & Boye (2016) modelled each possible landmark given by a person as a feature vector. The salience value for such a landmark is calculated as a weighted sum of these features. They propose the personal salience model for every participant can be built based on a landmark's position, type and context. They experimented with 10 participants with age averaging 27.3. The participants evaluated landmark positional features based on the distance and angle between the landmark and its selector.

If a specific landmark belongs to a particular type, they noted the value as 1 or 0; otherwise. The contextual features represent how many objects in candidate sets have the same value for all types of features. When building these models, they include different features (positional, type and texture features). For each feature ($x = (x1, \ldots, xn)$), a specific weight ($w = (w1, \ldots, wn)$) is assigned when computing the final salience value for a landmark. And the formula for computing salience value for each feature is a linear combination ($x * w$).

The focus of Götze & Boye (2016) is to derive a personalized model for each participant in predicting a potential candidate landmark for any new environment. Their model is innovative in predicting potential landmarks. However, some limitations are worth discussing. Their approach is akin to supervised learning, where the model must rely on carefully labelled data. When the problem deals with ambiguity and biases in human choices, such as choosing a restaurant as a landmark while another prefers the school, the possibilities for landmarks become endless. Thus, their model faces generalization issues. The method may give good performance for a map navigation application. For example, the Google map since users may require a customized service. But in robot navigation, a model for landmark selection should be general and robust to different machines.

2.3.3.3 Structural salience of landmarks for route directions

Xi et al. (2016) proposed a model to calculate salience information as shown in Equation (1):

$$S_s = S_v W_v + S_{se} W_{se} + S_{st} W_{st} \quad \text{..... (1)}$$

$$\text{Where } 1 = W_v + W_{se} + W_{st}$$

Equation (1) shows S_s as the final salience value for an object, S_v represents the visual salience while S_{se} is the semantic salience and S_{st} stands for the structural information. W_v, W_{se} and W_{st} are the weights assigned to visual salience, semantic salience and structural salience. They stated the weights could be fine-tuned to meet the requirement for different contexts. Different weights are necessary for structural salience, when S_{st} is calculated, the set of weight factors should comply with the structural taxonomy hierarchy as illustrated in Figure 2.8.

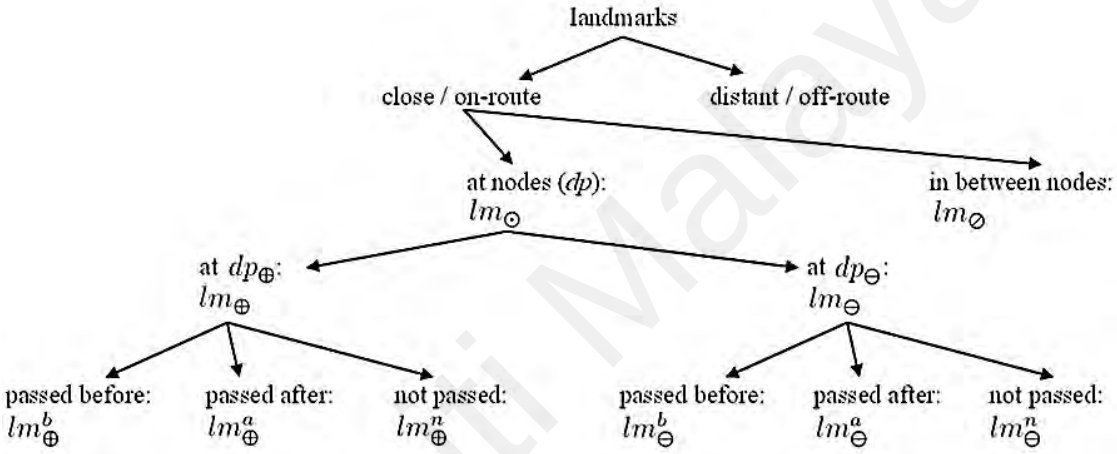


Figure 2.8: Hierarchy of landmark taxonomy (Xi et al., 2016)

2.3.3.4 Including landmarks in routing instructions

Duckham et al. (2010) introduced a novel model to examine if an object's category is suitable to be selected to be landmarks. They focused on categories rather than individuals because detailed information is usually hard to obtain, like colours and shapes. They used a heuristic approach and provided a series of criteria used by experts to mark suitability for a category chosen as landmarks. Table 2.2 demonstrates part of the visual factors assessed in their research.

Table 2.2: Visual factors (Duckham et al., 2010)

Character	Factor	Explanation
Visual	Physical size	Larger POIs are more easily seen and better candidate landmarks than smaller POIs.
	Prominence	POIs that are visually prominent (e.g., bear visible signs, markings, architecturally imposing) are better candidate landmarks than those with few or no distinguishing markings.
	Difference from surroundings	POIs that are typically different from their surroundings are preferable landmark candidates.
	Nighttime vs daytime salience	POIs that are highly visible both in day and night are better candidate landmarks in the context of the case study, since Whereis routing instructions may be printed out and later used during day or night
	Proximity to road	POIs that are closer to the road are more likely to be seen by navigators, and so are better candidate landmarks.

The abbreviation POI refers to the point of interest that can be considered an object category. They suggest that bigger objects that are visually prominent in both daytime and nighttime and close to the road are ideal candidate landmarks when selecting landmarks. The second character to select a better candidate landmark is called prominence. The term implies that a good landmark should be obvious and have distinctive markings (Duckham et al., 2010). For example, the Twin Towers is a classical prominence landmark. The building is well-known for its shape and height. Moreover, it is a pretty unique building compared to its surroundings.

The Leaning Tower of Pisa is also a good explanation of prominence. Typical buildings are upright. The amazing thing is that the main body of the Leaning Tower of Pisa is not vertical to the ground and has a certain incline angle. Furthermore, such a feature is memorable to people and architecturally imposing, making the tower a remarkable landmark. Difference from surroundings stands for how notable an object is or, in other words, being distinctive and unique. The nighttime vs daytime salience explains that a good candidate landmark should be visible during the day and night. The factor proximity to the road belongs to structural attention (Gangaputra, 2017).

2.3.3.5 Colour Salience

Different colour models such as RGB, HSV (or HSB) can compare colour features on images. However, not all of them are qualified for the real world since many factors in the physical world may cause colour differences (Szafir, 2017). Colour difference refers to the distance between the mathematical presentation of colours compared to human perception and an embedded camera.

Figure 2.9 shows that the shadowed area in the image is grey, similar to other image areas. However, from people's point of view, the shadow colour is close to black, different from grey. Indeed, when the image is transformed into valued pixels, the value will differ between the shadow and exposed areas. HSV, one of the popular colour models used in real life, will be ideal for dealing with such a problem due to its nature (see Figure 2.10).

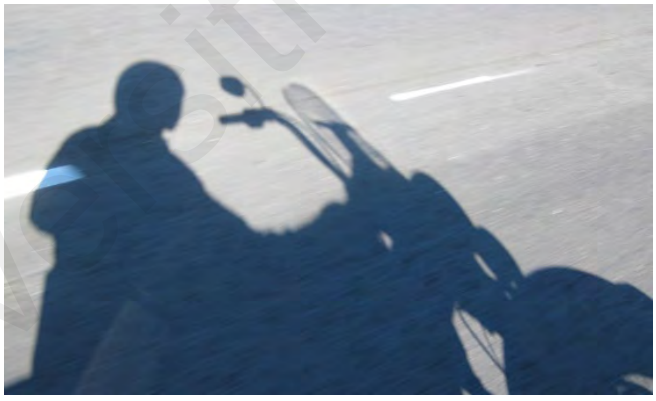


Figure 2.9: Sample of colour difference

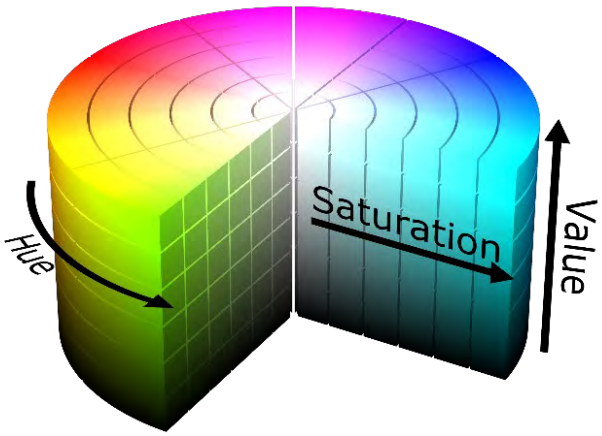


Figure 2.10: HSV model

HSV stands for three perceptual variables: hue, saturation and value or brightness, respectively. The model is based on human vision, making it suitable for cameras. Therefore, the value brightness becomes robust to this model. The shadow in Figure 2.9 will have the same colour when input to the computer. The only difference is the difference in illumination. Peter et al. (2010) proposed a model (Equation 2) for computing the colour salience where Sc is the colour salience and the hue, saturation, and value is assigned weights. The researchers proposed that hue plays the most significant factor with 75% weight compared to saturation and hue, which takes up 25% in getting colour salience. It is not reported why other weight combinations are not attempted.

$$S_c = 75\% * H + 20\% * S + 5\% * V \quad \text{..... (2)}$$

2.3.3.6 Shape Detection

Shape information is a crucial indicator in examining a landmark's visual appearance. Figure 2.11 shows a group of rectangles with one circle blended in. Although the rectangles are varied in size, the circle is distinctive; thus qualifies as a landmark in this example. A feature-based image retrieval method is called content-based image retrieval (CBIR). The main idea of the technique is to use image properties such as intensity value, colour and texture to retrieve objects in an image.

Glauco et al. (2011) developed a CBIR shape description system using salience points. According to them, shape saliency refers to the points that appear at the curve with a high curvature value and true corner points. Figure 2.12 demonstrates the idea of salience points on the shape of different objects.

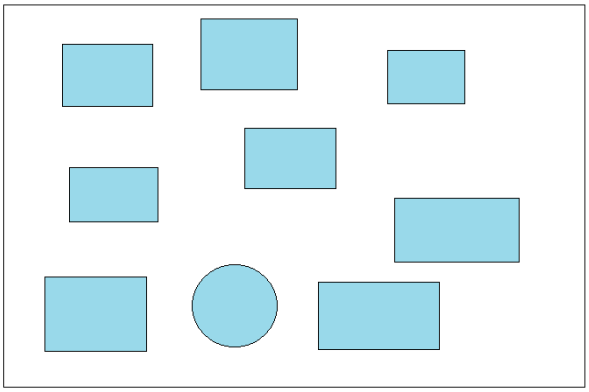


Figure 2.11: A group of rectangles with a circle

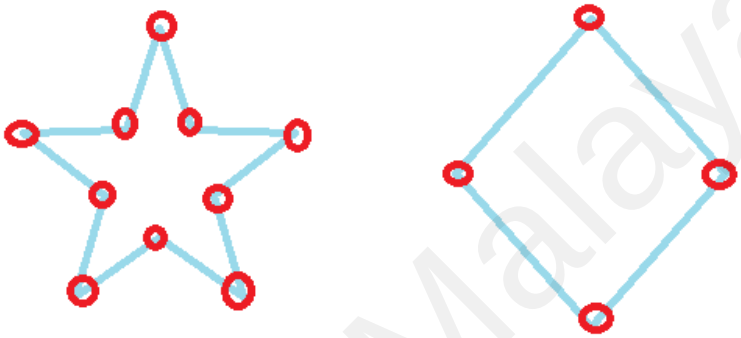


Figure 2.12: A representation of salience points on stars and diamond

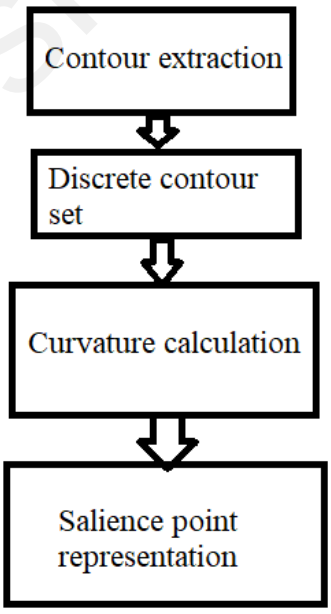


Figure 2.13: Flow chart of salience point calculation (Glauco et al., 2011)

The main idea in obtaining these salience points on a closed curve is to calculate the curvature value of the contour. A reasonable set of curvature threshold can be defined by set $A = \{p1, p2, p3 \dots\dots\dots, pn\}$, where A represents the discrete representation of the object contour, and set $V = \{V1, V2, V3 \dots\dots\dots, Vn\}$, where V stands for the curvature value set of the set A is produced. Some of the points, those with low curvature value, are excluded. The remaining points are the salience point of desire. Figure 2.13 shows the process of getting shape salience points.

Once the orthoimages are obtained, the shape of different buildings will then be possible extracted. Hence the shape salience of visual attractions can be calculated using a quantitative model. Equation (3) depicts the formula used to calculate the continuous curve curvature value:

$$C = \frac{x''y' - x'y''}{((x')^2 + (y')^2)^{\frac{3}{2}}} \quad \dots\dots\dots(3)$$

Where x' and y' , x'' and y'' represent the first and second derivative of a parameterized curve function. However, image data stored in machines are two-dimensional arrays, which is discrete data. And cannot usually be described as an entirety. Moreover, the dataset used to examine the algorithm proposed by Glauco et al. (2011) is continuous. It is recommended to focus on discrete data for image-based work since they are stored as discrete data in computers. The difference is that the curvature value calculated using discrete data is only approximation.

2.4 Image processing technique in landmark extraction

Image processing has several techniques for valuable information retrieval from images. Images can be obtained from various observing and capturing systems in different forms. Furthermore, the human eyes can directly react with image information and produce

visual perception. However, information is 3D in space in the real world, whereas it is generally 2D in the digital world (Zhang, 2017). A digital image stored in a computer is re-constructed as a 2D array $f(x, y)$ where x and y represent the x-axis and y-axis coordinate, and f is the value of the point (x, y). For example, in a grayscale image $f(x1, y1)$ is the intensity value of the point $(x1, y1)$ while for colour image (3-dimensional image), $f(x1, y1)$ in the 3D image refers to the colour value at point $(x1, y1)$ (Zhang, 2017).

2.4.1 Image filtering

Image filtering, also known as smoothing, is a technique to remove unwanted pixels and improve the quality of images (Chandel et al., 2013). This study reviewed different filtering methods categorized into linear and non-linear filtering. For linear filtering, the changes made to each pixel is an arithmetic operation where non-linear filtering contains logic operations like “complement” and “AND” computation (Zhang, 2017). Figure 2.14 visualizes the logic operation. Typically, linear filtering produces predictable results, while it is hard to imagine the output of non-linear smoothing.

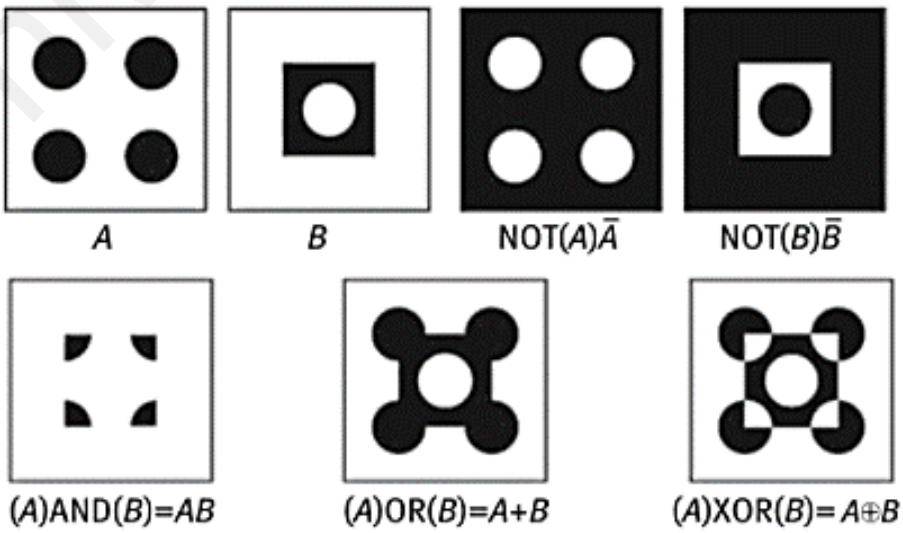


Figure 2.14: Samples of logic operation (Zhang, 2017)

Figure 2.15: Original image (left) and the same image smoothed by Butterworth filter (right). Taken from Zhang (2017).

Figure 2.15 shows an original image and the same image smoothed by a low-pass filter. The original image appears to be sharper to human eyes. However, in grayscale values, they share similar intensity values for each pixel (before and after filtering). The intensity value can range between 0 and 255. The higher the intensity value, the higher is the greyscale gradient for the pixel. When the pixel value is closer to 255, the colour of such pixel will be close to white. Pixels with lower intensity values will get darker and near black at 0. As the name suggests, a low-pass filter only allows a small greyscale value to pass, and a high pixel value will be filtered.

Due to the physical limitation, noises are very likely to arise in the transmission and capture process (Zhu & Huang, 2012). Noises are usually divided into three classes: Gaussian, impulse, and balance noise. Some noise has high-intensity values like snow noise, and some are dark. Picture details are usually found to be high-frequency components. Image filtering is a necessary step before further computation. If an image is only convoluted with a low pass filter to remove high noise, details of the image may lose, like the information for edges. An appropriate filter is critical in removing noise and not compromising detailed information simultaneously.

A bilateral filter (Song et al., 2014) is a non-linear filter that can smooth image noise without affecting the edge information. How bilateral filtering works is that when determining the new value for a specific pixel, the filter will consider the original and its neighbourhood values. If the result demonstrates the pixel is an identically distributed random point or independent of its neighbour, the pixel will be filtered and considered noise (Do et al., 2011).

2.4.2 Image Segmentation

Image segmentation is one of the most critical steps in image processing. The result of this phase directly affects following operations such as morphology, feature extraction and other image processing computations (Sharif et al., 2012). An image is a form of media that contains much helpful information. People will get valuable and meaningless information from different images. Getting meaningful information and ignoring unnecessary data is a crucial point in image processing. To understand the information carried by an image, the first thing to do is to segment images. In practice, not all part of a picture is valuable; the attention is paid mainly on the areas with specific characteristics (Yuheng et al., 2017).

Image segmentation means portioning an image into several parts or several sets of pixels (super pixels). It is commonly used in finding the contours or location of an object (Singh et al., 2010). To be more specific, pixels sharing certain visual characteristics will be labelled the same, and image segmentation is the process of assigning every pixel of an image with a label based on the previously mentioned rule. The image segmentation generates several labelled areas where pixels belonging to the same area are homogenous (Narkhede, 2013).

Partitions after the process are objects that have similar visual properties such as colour, intensity value or texture (Senthilkumaran et al., 2009). Image segmentation techniques are commonly used in medical areas like blood cell segmentation (Li et al., 2016, Nee et al., 2012, Tomari et al., 2014). In their blood cell segmentation studies, the main object is identifying red blood cells and white blood cells. And the method used is image segmentation approaches such as threshold method, artificial neural network, k-means clustering and many more.

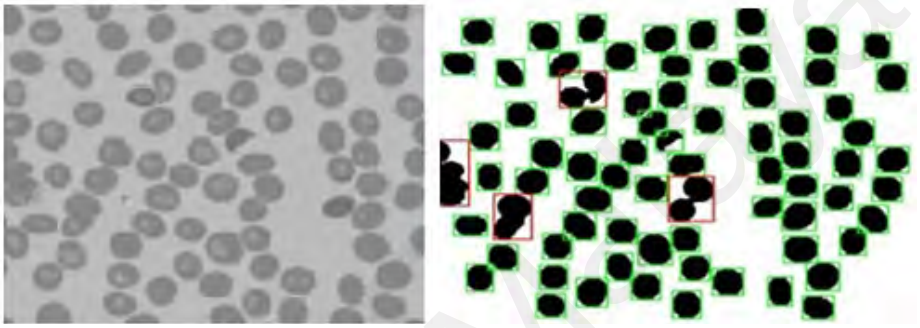


Figure 2.16: An example of the blood cell segmentation process with the original (left) and segmented image (right). Taken from (Tomari et al., 2014)

Figure above illustrates that images with many objects can be successfully segmented, and even the contours of each object are successfully recognized by image segmentation. Saini et al. (2014) classify the image segmentation method into two categories. One is edge-based image segmentation, and another is region-based image segmentation. Image segmentation methods using discontinuity belong to the boundary-based approach, and methods using similarity characteristics are classified into the region-based approach.

Jeevitha et al. (2020) divided those image segmentation techniques into five categories. Thresholding based, edge detection based, region based, feature based clustering and neural-network based segmentation. The thresholding-based method is widely used in computer vision, the idea is to find a suitable threshold to divide pixels of an image into two categories (foreground and background). The main idea for edge detection based is to find the significant changes in intensity values, such as edges. The third category is to

divide an image into regions based on the properties predefined. The clustering-based methods such as k-means are also widely used while the neural-network based method take advantages of power computer to train models to predicate each pixels of an image.

In this dissertation, the ideas of threshold-based, edge-based and region-based methods are considered to be used in the dissertation, and following are the details.

Al-Amri et al. (2010) researched different image segmentation techniques. Their work focuses more on the first category, namely the histogram threshold approach. Image histogram (Sutton, 2016) is a type of histogram that shows the frequency of occurring pixel values of a whole image.

The image histogram is denoted as $P(r_k) = \frac{N_r}{N}$. To explain the equation, r_k is the intensity level and N_r is the number of pixels that has an intensity value of r_k while N is the total number of pixels (Bora, 2017). The equation accurately illustrates the actual meaning of the image histogram. An image segmentation based on histogram thresholding uses the equation to cut an image apart.

Al-Amri et al. (2010) grouped image histogram thresholding techniques into three classes, the local, global, and the split, merge and growing techniques. The local approach depends heavily on the properties of neighbourhood pixels. The global method considers all pixels using global histogram properties, while the split, merge, and growing techniques use the similarities and geometrical proximity to get a good segmentation result. The thresholding method works by using an appropriate threshold value T calculated by algorithms. The foreground and background information will be

discriminated by *T*, and the target image will be transformed into a binary image where the pixel values are only two values, either 1 or 0.

Binary images are considered a special kind of greyscale image. Usually, a greyscale image has a greyscale level ranging from 0 to 255, i.e., 256 level total. If a greyscale image contains only two intensities, 0 and 255, it can also be considered binary. The advantage of converting images into binary images is that the complexity can be reduced and increase computation efficiency (Al-Amri et al., 2010). The ideal image is that when the histogram only has two peaks, the best threshold value will be the middle value of the peaks.

The most popular automatic threshold image segmentation technique is Otsu's method. The Otsu method considers the most significant interclass variance, which maximizes the variance value of interclass separated by selecting a globally optimal threshold value. The calculation is fast and straightforward, and the segmented effect is noticeable when foreground and background have high contrast (Yuheng et al., 2017).

Edge-based image segmentation, also known as the boundary-based method, detects significant changes in intensity value. Edges are a sign of discontinuity or contour endings (Zaitoun et al., 2015). And edges are often found at the boundary of two regions (Sharma et al., 2012). This kind of method is usually applied to greyscale images. And it is meaningful to find significant discontinuities in the grey level. Features are extracted around edge areas such as corners, curves, straight lines. According to Saini et al. (2014), all edge-based methods are categorized under 1st order and 2nd order derivatives.

Table 2.3 shows the image segmentation methods under each derivative. The approaches in the table used different operators to detect edges of objects in images. Although they are categorized into different classes, the central idea is the same. They compare two adjacent pixel values to judge whether these two adjacent pixels appear on the edge area.

Table 2.3: Edge-based image segmentation methods (Saini et al., 2014)

1st order Derivative	**2nd order Derivative**
Prewitt operator	Laplacian operator
Sobel operator	Zero-crossings.
Canny operator	
Test operator	

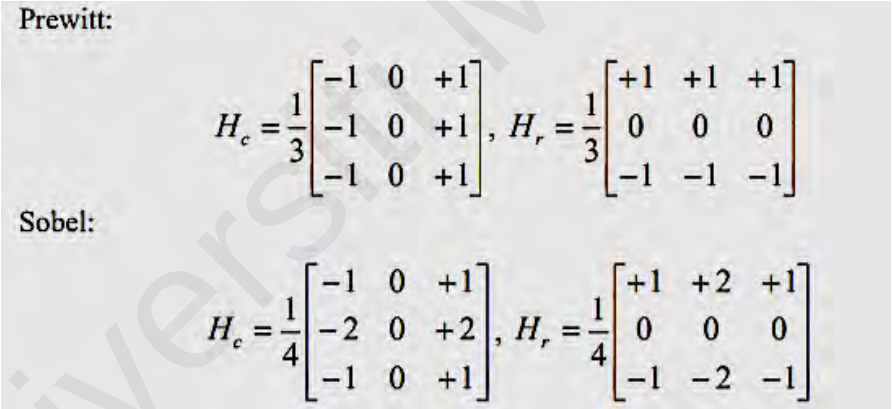


Figure 2.17: Prewitt and Sobel operator for edge detection (Oskoei, 2010)

Figure above shows the example Sobel and Prewitt operator for the 1st order derivative edge detection method. The Prewitt and Sobel operators both follow the same pattern. The detection is based on columns and rows of images. For example, Sobel H_c detects a vertical edge between two objects. The pixel value around this area will have an abrupt change if the vertical edge appears. Calculating H_c using a right-side convoluted value minus a left-side convoluted value will show a significant difference.

If there is no edge, the left-side and right-side parts will have the same pixel values, so the difference is 0. For example, a pure black 3x3 image with a value of 0 across the pixels gets a 0 difference when taking the right-most column minus the left-most column. If the 3x3 image is half back and half white, with a clear vertical edge in the middle, the pixel values for the right-side part are all 255, and 0 for the left-side. If the image uses H_c to detect edges, the result will be $(255x1 - 0x0)\ x\ 3\ =\ 765$. In summary, the edge detection approach is based on grey level changes. The approach generates a good result whenever pixel values between edges have significant change.

For region-based segmentation, a region of an image is a connected homogenous subset of the whole image. The pixels in the same region share identical characteristics like intensity value or texture (Narkhede, 2013). This method is more straightforward than the edge-based method and, to some extent, immune to noise (Kang et al., 2009, Zhang et al., 2008). For region-based methods, the pixels are assigned to different regions or objects according to pre-defined criteria. The region-based methods are more robust to noise and are based on similarities between pixels. On the other hand, the edge-based methods cluster partitions according to sharp intensity changes (Kaganami et al., 2009). Table 2.4 demonstrates the differences between the two categories:

Table 2.4: Differences between edge-based and region-based methods. Extracted from Kaganami et al., 2009

Region-based segmentation	**Edge detection**
Closed boundaries	Boundaries formed not necessarily closed
Multi-spectral images improve segmentation	No significant improvement for multi-spectral images
Computation based on similarity	Computation based on a difference

Region-based image segmentation also requires thresholding methods—the main idea including pixel value similarity and spatial proximity like the Euclidean distance. Commonly used region-based methods are region growing, region splitting and merging and watershed transformation. The region growing method starts with a specific pixel and examines its neighbourhood based on similarities. If its neighbourhood meets the similarity requirement (for example, connectivity), adjacent pixels will be added to the same group as the original pixel. The algorithm then chooses the adjacent pixel next to the first one, and the process is iterated until all pixels are reviewed.

The advantage of the region growing method is that all connected pixels are guaranteed to be in the same group. However, when an image is compromised by noise, the result might not be ideal (Zaitoun et al., 2015). The region splitting and merging approach is the opposite of region growing. The splitting and merging method regards an image as a superset and divides it into several subsets. After splitting, adjacent subsets will be merged if the variance is slight. The process is ended when there is no more splitting and merging required. Another advantage is the splitting and merging method has no manual iteration. However, the input image should be formatted into a pyramidal grid structure. The watershed method will transform images into gradient images (Saini et al., 2014) and consider grey values as the surface elevation. Then the water flows out from the lowest grey value. If the flood crosses two converges, then a dam is built to indicate a boundary between them.

2.4.3 Morphology

Morphology operations are also used widely in image processing. Common geometric transformations such as rotation, colour correction and rotation, etc., are essential in industry production. Mathematical morphology is also an important concept in computer

vision. Techniques of mathematical morphology target to process shapes and geometric information (Kaur et al., 2013). Sharp intensity value changes often reflect discontinuities in-depth, surface orientation, illumination changes, and material changes (Bai 2010).

In other words, when two pixels have a significant gap of pixel value, such pixels are usually border pixels on an image. Tambe et al. (2013) mentioned that mathematical morphology plays a vital role in image shape obtaining. There are four operations in mathematical morphology, which are dilation, erosion, opening and closing. Opening and closing operations are derived from dilation and erosion, where opening erodes images first and then dilates while closing operation doing a reverse order. The basics are dilation and erosion operations, and they are commonly defined for sets first. Dilation will expand a set while erosion shrinks a set.

The mathematical symbol for dilation (Tambe et al., 2013) is denoted as:

$$X \oplus B = X + b = \{x + b : x \in X \,\&\, b \in B\}$$

Where X stands for a shape and B represents the structural element used to dilate image X. The output generated by this operation is a set of translated points that structural element B has a non-empty intersection with X. How this works is like the process of image filtering, the difference is that when smoothing an image using linear filtering, a kernel is used to generate dot products while dilation structural element B is doing a logical operation. If the centre point of B overlap with a point of the image that has shape X and at the same time structural element B hit at least one point of shape X, then such point is verified and will appear at the dilation result image.

A simple application for dilation is to fill gaps (Tambe et al., 2013). If a small gap separates a large object from an image because of noise or artistic design, then such an object may be considered several small objects situated close to each other. However, there should be only one large object. Thus, if a proper structural element dilates such an object, the gaps between parts will be filled, and discontinuity will disappear, improving the recognition accuracy.

The mathematical symbol for erosion (Tambe et al., 2013) is denoted as:

$$X \ominus B = X - B = \{ re : (B + re) \subseteq X \}$$

Where X stands for a shape and B represents the structural element used to dilate image X. The eroded image is a point set that contains points when structural element B's centre overlap with any of these points on the original image X, and every pixel is covered by structural element B. This process is similar to dilation. The structural element B at this time also performs a logical operation.

The difference between dilation and erosion is that erosion operation works like "AND" logical calculation. Only if all pixel value covered by B is equal to 1, then such point covered by B's centre point is considered to be a valid output point and will be added to the output image "re". At the same time, dilation works more like an "OR" operation. If at least one pixel is 1, the point covered by B is valid.

The erosion process, to some extent, can be used to remove positive impulses and filter some noise (Kaur et al., 2013). By introducing dilation and erosion, gap issues can be improved, and some slight noise can be removed. The closing operation removes negative impulses and preserves positive ones, a typical image processing task. The structural

element can be customized, and a proper structural element can determine how the image will be exactly dilated or eroded (Raid et al., 2014). Chudasama et al. (2015) stated that a structural element would be applied to all possible locations, and this operation will generate a new binary image (images only contain pixel values 1 and 0). Moreover, a structural element can be diamond-shaped or look like a square. Figure 2.17 shows the original and output images eroded and dilated by a structural element (Sreedhar et al., 2012).

Figure 2.16 shows the original image's mathematical morphology background detection, erosion, and dilation operations. The figure shows that the flowers after dilation have richer petals and thicker edges than after erosion. Dilation morphology adds pixels to the boundaries of original images, while erosion reduces some details. Goyal (2011) provide an intuitive illustration of how erosion works, offering an in-depth understanding of mathematical morphology. Furthermore, in her work, an edge extraction algorithm using erosion has been reviewed.

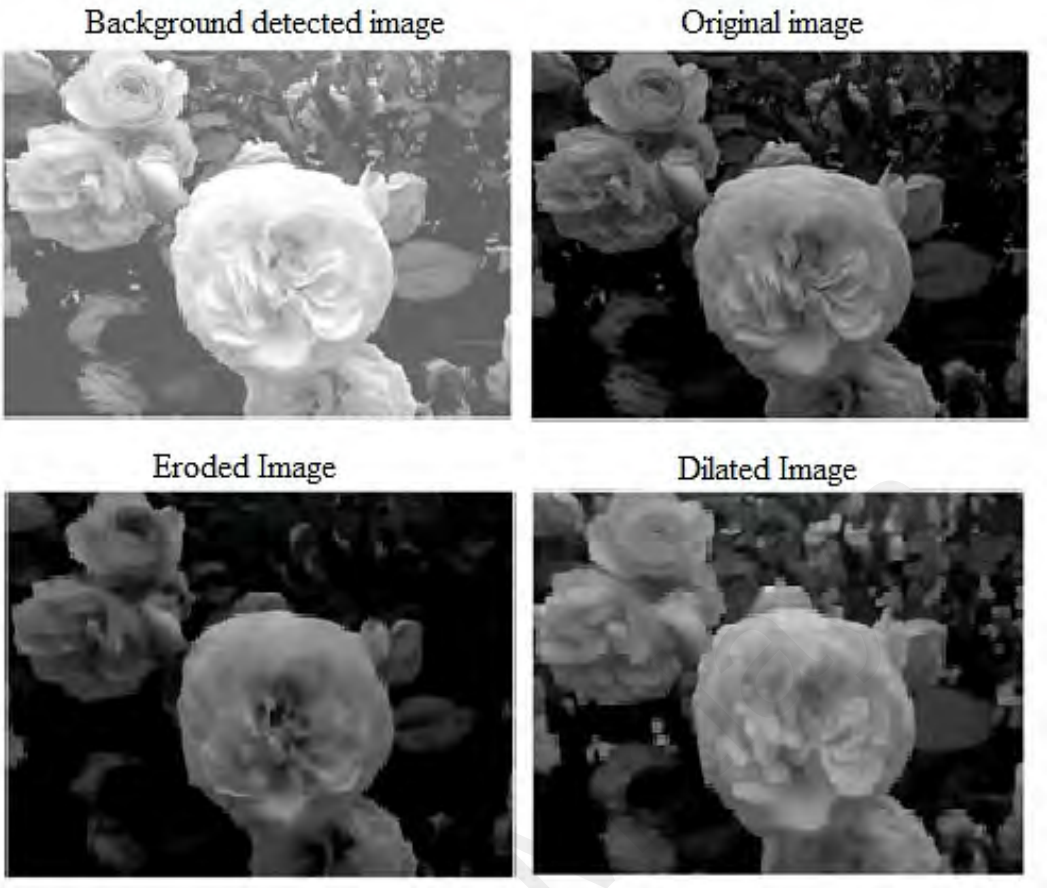


Figure 2.18: Mathematical morphology background detection, erosion and dilation operations on an original image (Sreedhar et al., 2012)

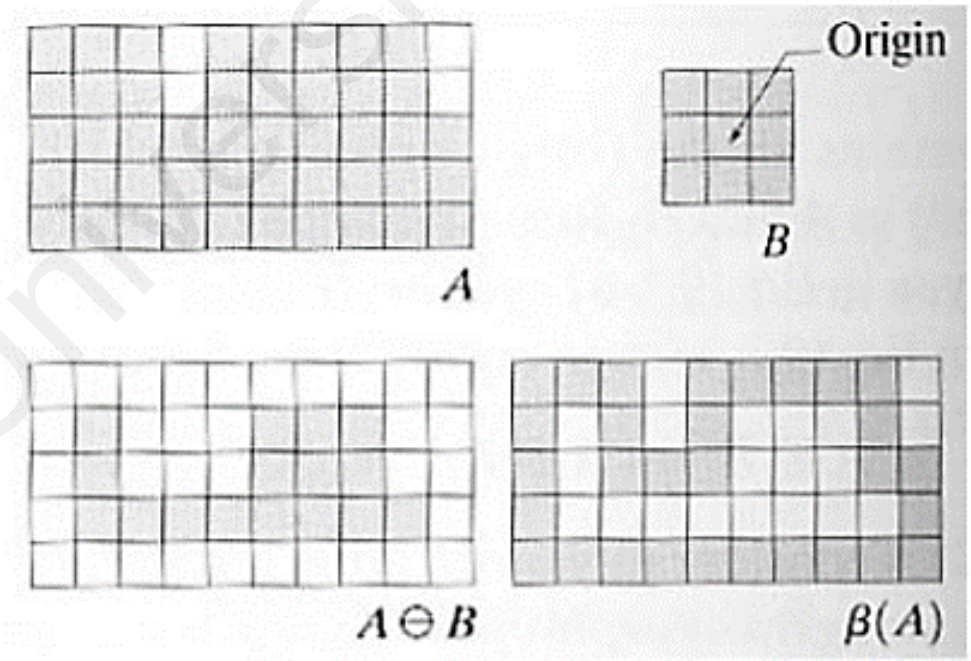


Figure 2.19: Edge extraction algorithm (Goyal, 2011)

In Figure 2.17, A represents the shape waiting to be eroded. B is the structural element used to erode shape A. The edges of A using this approach is denoted by: $\beta(A) = A - (A\Theta B)$. The centre point of structural element B is where the arrow points for image erosion. In this case, only the 8-neighbourhood pixels covered by structural element B are considered valid since the structural element B is a 3x3 square. This process ensures that the valid pixels must be inside shape A, and using original shape A minus the internal pixels. The result will be edge points of shape A.

2.5 Summary

This chapter begins by highlighting the role of a landmark in human and robot navigation. The theory behind computing the visual salience metric for landmarks is presented. The discussion covers how information about visual appearance, namely, object size, colour and shape, influences the object salience value. Eliciting visual landmarks requires image processing. Thus, the chapter continues with traditional image processing methods, including image filtering, foreground and background image separation, morphology operation and image segmentation analysis. The following chapter presents the methodology adopted in this dissertation.

CHAPTER 3: METHODOLOGY

3.1 Overview

This chapter presents the methodology adopted to address the dissertation objectives. Objective 1 requires methods in segmenting urban landmarks from an image, while Objective 2 deals with developing an algorithm to quantify visual salience indicators for urban landmarks extraction.

3.2 Algorithm design

The algorithm has three modules: pre-processing module, salience calculation algorithm and landmark output. The pre-processing module aims to smooth the input images and produce an image with noise removed. The salience calculation module is the core part of the whole algorithm. The salience value for potential candidates is figured out using the output fine-tuned images from the first module. The best salience value among the landmark candidates finalizes the landmark for robot navigation. The open-source computer vision library (Open CV 3.4.2) and Python programming language (Python 3.7.1) are used for algorithm development. Figure 3.1 shows the four stages of the algorithm proposed.

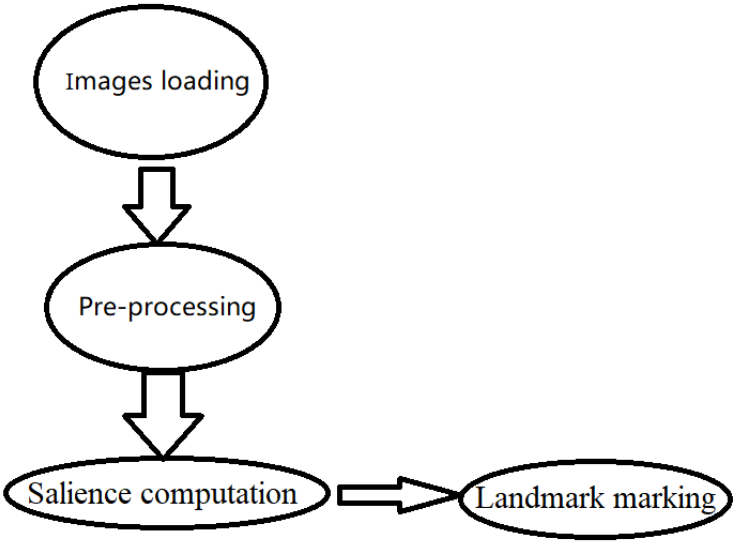


Figure 3.1: The four stages of the algorithm

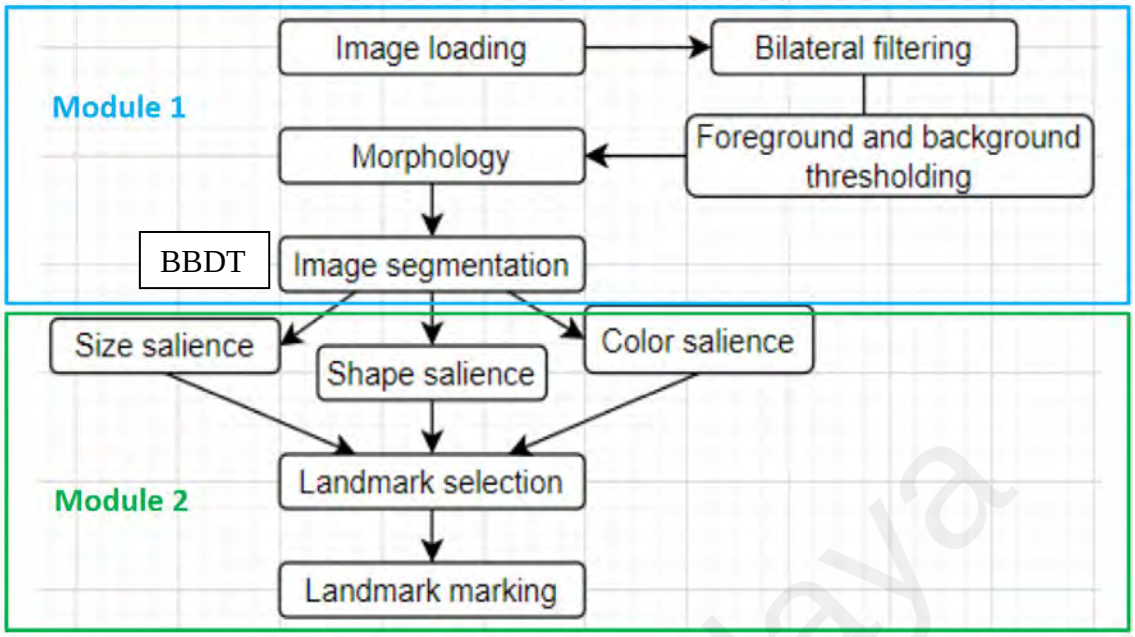


Figure 3.2: Algorithm design divided into two main modules

Phase	Step	Applied Techniques
0	Step 1	Load the input images (frames) from disk.
1	Step 2	Applied gaussian blur technique to filter the images.
2	Step 3	Threshold the image to get foreground and background divided and generate binary image
3	Step 4	Morphology calculation(Close operation is used in this algorithm)
4	Step 5	Perform image segmentation. BBDT algorithm for 8-way connectivity is applied.
5	Step 6	Perform area salience calculation
6	Step 7	Perform color salience calculation
7	Step 8	Perform shape salience calculation
8	Step 9	Compare salience of each candidate landmark
9	Step 10	Get the final results and draw contour

Figure 3.3: Steps involved in the proposed algorithm

Figure 3.2 shows the proposed algorithm design dividing the four stages into two modules. Module 1 addresses Objective 1 regarding image segmentation using block-based connected components labeling with decision trees (BBDT), whereas Module 2 addresses Objective 2 regarding the visual salience indicators extraction. Figure 3.3 shows the overall algorithm steps from start to finish.

3.3 Data acquisition

The data here is defined as aerial imaging of urban landmarks in the surrounding environment. Aerial imaging of urban landmarks can come from many sources. In this dissertation, the dataset is built from different sources. The first is aerial images downloaded from online repositories, and the second is live video clips collected using micro aerial vehicles or drones. Aerial images or video clips are selected for model development initially as the overall shape of buildings is more distinct from birds-eye-view. Images and video clips from frontal view will be collected and examined to extend the visual salience model. Moreover, images collected from internet sources such as Google, Bing, and other picture sharing websites will test the algorithm. The image resolutions vary regardless of their source.

3.4 Module 1: an algorithm for landmark segmentation

This section describes the model development from stages in pre-processing, the salience calculation, landmark marking, and the considerations for the algorithm output design. The CPU used is Intel(R) Core(TM) i5-8300H CPU at 2.30GHz speed with 16GB RAM. Windows 10 is the operating system, and the Python environment is Sublime Text3.

The pre-processing module includes the basic image processing operations on the data acquired for the algorithm development. First, the images will be smoothed using a

Gaussian filter to remove unwanted noise. Then, the colour model will be changed from RGB (default) to HSV. This step is essential to make sure the colour is preserved and reduce the illuminance impact. A pixel operation is applied next, specifically in detecting the edges of each object and performing segmentation.

3.4.1 Image loading

The first step is to load images with the CV.READ function as follows:

```
26        im=cv2.imread(imgPath)
```

Table 3.1 shows the supported image extensions for the CV.READ function. The study selects JPEG type files since the extension is standard on cameras, including drones and mobile robots.

Table 3.1: Supported Input Images

Image Types	Extensions
Windows bitmaps	*.bmp, *.dib
JPEG files	*.jpeg, *.jpg, *.jpe
JPEG 2000 files	*.jp2
Portable Network Graphics	*.png
WebP	*.webp
Portable image format	*.pbm, *.pgm, *.ppm *.pxm, *.pnm
Sun rasters	*.sr, *.ras
TIFF files	*.tiff, *.tif
OpenEXR Image files	*.exr
Radiance HDR	*.hdr, *.pic
Raster and Vector geospatial data supported by Gdal	N/A

3.4.2 Gaussian and bilateral filtering

Commonly, natural images contain noise. Sometimes noises are harmless to photography. However, it is a different story when it comes to image processing. Noises will drastically influent the result for salience calculation. The salience calculation for the study is calculated from three aspects, size, colour and shape. All of the calculation is based on pixel manipulation, that is, every pixel will be included in computation process even if it is a noise point (without filtering). If there are many noise spots on a picture, the algorithm for the salience calculation module will count those noise all. Besides, suppose noises are not removed before further processing. In that case, the module will consider noises as potential landmarks, and as a result, an extra calculation will be needed, thus undermining the efficiency of the whole algorithm. An appropriate de-noise method should be considered in tackling this problem.

Currently, there are many approaches to help remove unwanted noises from images. Filtering the source image with a specific kernel is one of the popular ways. Kernels such as adaptive median, median, average, and Gaussian filters are commonly used. Although the filters mentioned above are all capable of filtering noises out, it would be a different story if those filters were applied to the wrong type of images. Different filters will perform differently according to the features of the images. For example, the salt and pepper noise is a type of noise present sparsely on an image with white and black pixels. The salt and pepper noise is supposed to be removed more efficiently using the average filter (Nader et al., 2017), while noise that appears following a normal distribution is more suitable using a Gaussian filter.

For images used in this dissertation, the salt and pepper noise, which usually appears during signal transfer over cables, will not be considered since a camera takes the picture. Instead, the noise is more likely to be Gaussian (noise complying with gaussian distribution). Figure 3.4 shows the classic Gaussian filter kernel, a 5x5 kernel that considers only the space domain distribution. In other words, no matter what type of pixels are, they are treated the same way. The Gaussian filter kernel may cause trouble to shape salience detection since contour information for landmarks may be blurred and lost, thus negatively impacting salience calculation.

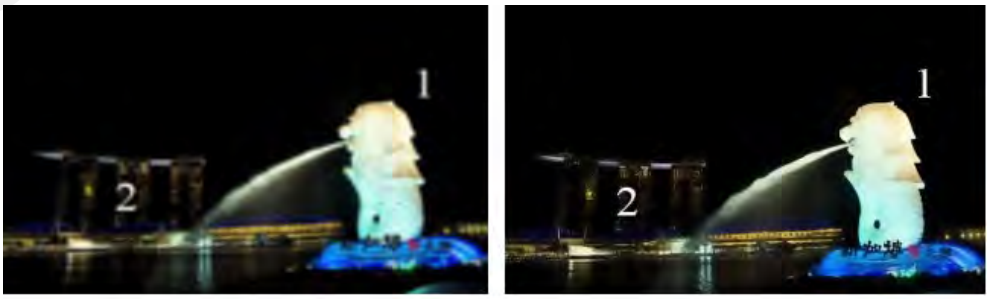
$$\frac{1}{273}\begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

5x5 example

Figure 3.4: Typical gaussian filter (5X5) sample

```
31      #Gaussian Filtering
32      if(myFilterMode == filterMode.gaussianFilter.value):
33          gaussianResult = cv2.GaussianBlur(im,(15,15),1.5)
34          # print("Gaussian is called")
35      elif(myFilterMode == filterMode.bilateralFilter.value):
36          #Gaussian double filtering
37          gaussianResult = cv2.bilateralFilter(im, 0, 100, 5)
```

Figure 3.5: Codes for filtering



Gaussian filter result Bilateral filter result

Figure 3.6: The Gaussian and bilateral filtering on the same image

Contour information describes the shapes of objects. Pixels that belong to edges often have sharp changes in pixel values. An improved version of the gaussian filter named bilateral filter is selected to keep such information and remove Gaussian noise. A bilateral filter has another additional weight parameter that indicates the similarity of adjacent pixels compared to the Gaussian filter, a bilateral filter will not only consider the space information like Gaussian filter but also introduce a factor that consider intensity values of a pixel's neighbors. Figure 3.5 shows the codes used in the dissertation for filtering. Figure 3.6 shows an image filtered by the Gaussian and bilateral filters. In the figure, the left-side image is processed with a Gaussian filter, and the right-side image uses the bilateral filter. The image filtered by the bilateral filter shows sharper edges. A bilateral filter makes objects' contours sharper when reducing the noises of images. In this dissertation, the proposed algorithm includes a bilateral filter in removing unwanted pixels and retaining edges details.

3.4.3 Foreground and background separation

After the image is denoised, the next step is to sift the information desired. An image is rich in information. The foreground represents the part of the image closest to the camera. Technically, the foreground contains useful information of the image. The image processing can detect changes in the image sequence to separate the foreground and background information using a calculated threshold. If a pixel value is greater than or equal to the threshold, it is set to 255. Otherwise, the pixel value will become 0. This operation is a prerequisite for the following steps, especially for the morphology and the connect domain calculation. Converting images to binary type is time-saving for other processes.

Thresholding selection influence the foreground and background separation. There are two ways to set up a threshold, either by hardcoding or automatically producing a threshold. Otsu's method is an excellent example of automatic thresholding. Otsu's method is a highly successful threshold generator that maximises the values between-class variance. The method processes image histogram, segmenting the objects by minimizing the variance of each class. Such processing works best for bimodal images as their histogram clearly expresses two peaks. Technically, Otsu's method assumes the threshold can divide an image with global mean intensity value (Mg) into two classes; TH1 and TH2. The TH1 represents the class whose value is greater than the threshold value and has a mean intensity value $M1$ with possibility $P1$. The TH2 is the class whose value is less than the threshold value and has a mean intensity value $M2$ with possibility $P2$. Equation (4) and (5) shows the formula:

$$\mathrm{Mg} = \mathrm{M1} * \mathrm{P1} + \mathrm{M2} * \mathrm{P2} \quad (4)$$

$$1 = P1 + P2 \quad (5)$$

The basic idea of Otsu's method is to select a threshold that can make maximum inter-class variance (TH1 and TH2). Equation (6) shows the derivation:

$$\sigma^2 = P1P2(M1 - M2)^2 \quad (6)$$

Otsu's method is highly successful as a global thresholding method, particularly for bimodal images. However, it has limitations when the object area is small compared to the background area, and the histogram no longer exhibits bimodality. Also, the foreground and background images are segmented based on the intensity value. So when the foreground and background intensities broadly vary compared to the mean difference, the histogram valley loses its peaks. The same outcome can appear when the image is

severely corrupted by additive noise. Despite its widely successful usage, Otsu's method can still contribute to segmentation errors with incorrect threshold selection.

This dissertation aims to apply the algorithm developed on real robots. Data acquired by robots are often noisy. When the robot is a drone flying from a vantage point of view, the urban landmarks can appear small compared to the background. For this reason, this dissertation explores both hardcoded and automatic thresholding generation methods for foreground and background separation. Users can decide which thresholding method to use based on their situation. Otsu's method is available for automatic thresholding, or users can handpick a thresholding value. Selecting a value for the self-designed thresholding has been a try-and-error approach. Over 30 non-bimodal images of the environment containing urban landmarks are used. The images were either downloaded from the internet or taken using a drone.

Figure 3.7 shows an image processed by Otsu's on the left-side and the hardcoded method. The Otsu's selected 99 for the threshold, while threshold 180 is fixed for the self-designed method. Otsu's method performs better as the image features a distinct foreground object. Figure 3.8 replicates the experiment on a non-bimodal image. The image used is an aerial view of an urban neighbourhood landscape. The drawing is done from a particular perspective, so the buildings appear small and absorbed by the background. In this example, the self-designed thresholding shows a better foreground separation than the background. Even to human eyes, it becomes easier to separate individual objects. The salient visual metric requires evaluation of individual landmarks and this self-designed thresholding supports the pre-processing stage.

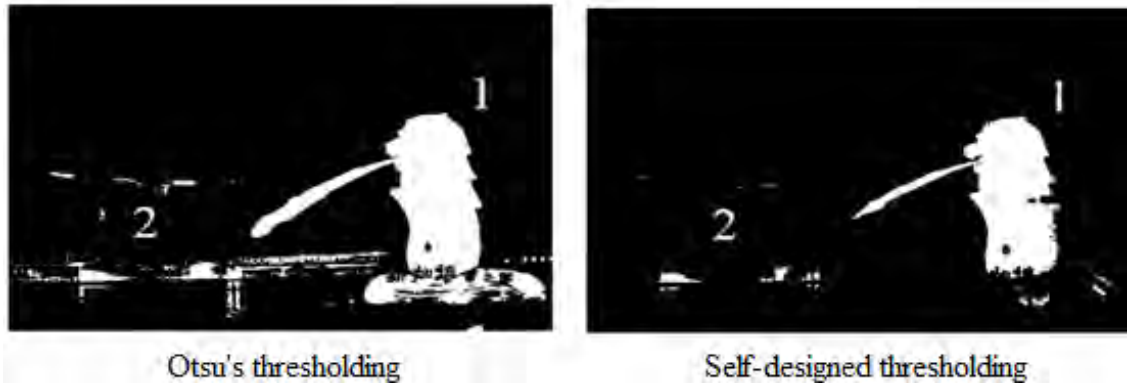


Figure 3.7: Bimodal image thresholding

Figure 3.8: Non-bimodal image thresholding

```
48      if(myThreshMode == threshMode.selfDesigned.value):
49          #for images containing many objects(pixel value variance is high )
50          _,th = cv2.threshold(im_gray,threshholdValue,255,cv2.THRESH_BINARY)
51          threhPicTitle = "selfThresh"
52          print("myThresh: "+ str(threshholdValue))
53      elif(myThreshMode == threshMode.otsuMethod.value):
54          # Otsu's method for images containing few objects.(Pixel value variance is low, likely to appear bimodal
55          otsu_value,th = cv2.threshold(im_gray, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)
56          threhPicTitle = "otsuThresh"
57          print("otsu: " + str(otsu_value))
```

Figure 3.9: Codes for foreground and background separation

Figure 3.9 shows the foreground and background separation codes that consider both the Otsu's and self-designed thresholding. Whenever non-bimodal or noisy images are encountered, the pre-processing set 180 (can be changed, the assumption is that object will have a high intensity value) as the threshold value rather than using the threshold value calculated by Otsu's method. Images with many objects are the typical signature of non-bimodal images. The pattern is familiar in urban landmark imaging taken from an aerial perspective, which is the use case in this dissertation. Only pixels with a threshold value greater than the self-designed thresholding will pass line 48 in the codes.

Otherwise, the image will go through Otsu's thresholding. Bimodal images typically feature fewer objects; the best case for Otsu's method is that there is only one big object in the image. With how data is acquired in this dissertation, getting images with a single large object is improbable. However, Otsu's method is still favoured for images with few objects, where each object are big enough.

3.4.4 Morphology operation

The morphology operation mainly fills the gap between possible connected elements, reducing the mistake caused by discontinuity. Discontinuity often occurs after the foreground and background separation. Figure 3.10 is the original image of the urban development landscape used in the previous section. The drawing is a good representation of an image of the city. Buildings are clustered; some have similar shapes and sizes, while others do not. Green is everywhere, depicting the background. Additionally, the landscape includes lakes, a good example of a nature-type landmark. Figure 3.12 shows the morphology operation result following the codes in Figure 3.11.

The morphology operation aims to separate foreground objects by visually labelling them according to RGB colours. Except for one, every building in Figure 3.10 is identified as a different object in Figure 3.12. See circle marking in the image before morphology operation. This happened here because discontinuity splits the same building into two separate units. The split leads to object miscounting and may cause salient visual metric problems. The morphology proposes a closing operation on separated objects.

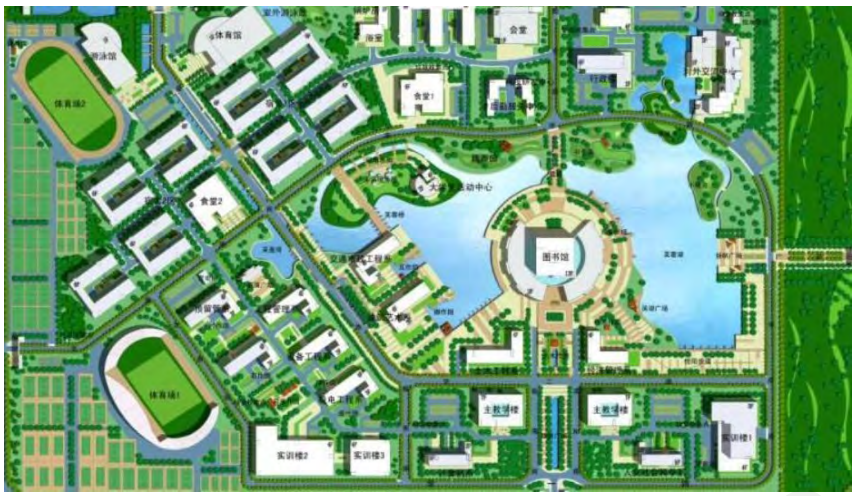


Figure 3.10: An urban development landscape blueprint

```
78      #structural element
79      kernel = cv2.getStructuringElement(cv2.MORPH_RECT,(3, 3))
80
81      #close operation
82      closed = cv2.morphologyEx(th, cv2.MORPH_CLOSE, kernel)
```

Figure 3.11: Codes for the morphology operation

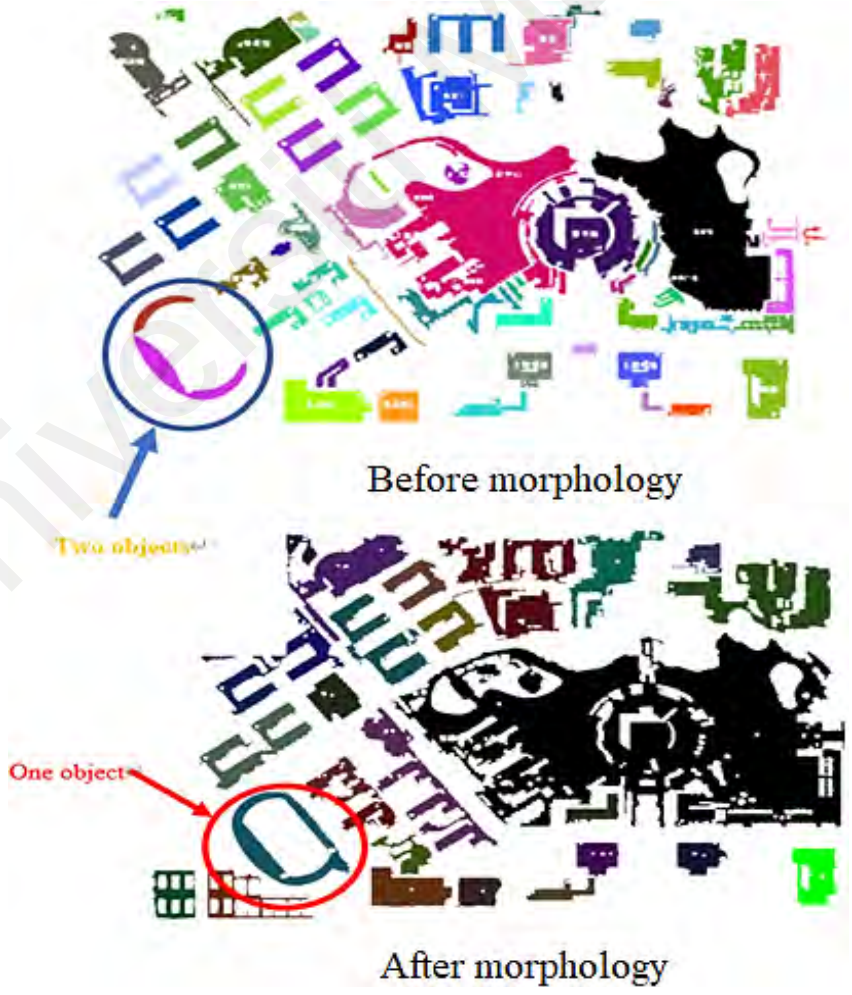


Figure 3.12: Example of using morphology in resolving disconnected object

The close operation contains image dilation and erosion, calculated using the same kernel. Figure 3.13 shows a kernel example. The geometric meaning of close operation is to connect two adjacent pixels. While dilation connects two adjacent pixels, erosion deletes the pixels that might not be desired. Moreover, the resulting image will be processed by the other parts of the algorithm.

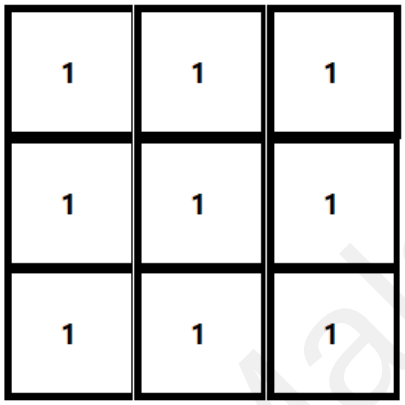


Figure 3.13: Kernel for a close morphology operation

3.4.5 Region-based Segmentation

The previous section shows how the morphology operation closes object discontinuity problems. Once object connectedness is no longer an issue, the pre-processing can determine the landmark candidates. The rule of thumb assumes that a potential landmark should be visually visible and remain whole as a connected entity. Therefore, every single object extracted after morphology can be considered a landmark candidate.

The region-segmentation process aims to separate the landmark candidates into individual pixel systems. Each landmark has its coordinates based on its pixel position on the image. The morphology operation significantly reduces miscounting possibility, so the number of systems makes up the number of landmark candidates in the image. A segmentation process is a region-based approach; pixels within the same area should have similar features. Images are converted into binary during the foreground and background

separation before passing to the image segmentation module. The pixel values are 1 for foreground and 0 for background. Segmentation by threshold will not work in binary mode. Therefore, the algorithm must rely on spatial or connectedness information to segment objects in the image.

A connected domain is defined as adjacent pixels within a connected domain with no gaps between adjacent pixels. Figure 3.14 shows two connected domains, a 4-connectivity or an 8-connectivity. The 4-connectivity and 8-connectivity have the same index in the middle, but their edges are configured differently. The 8-connectivity counts more pixels, which is adopted by this algorithm. The reason is that sometimes, even the input graph is morphed. Sometimes, discontinuity still may happen.

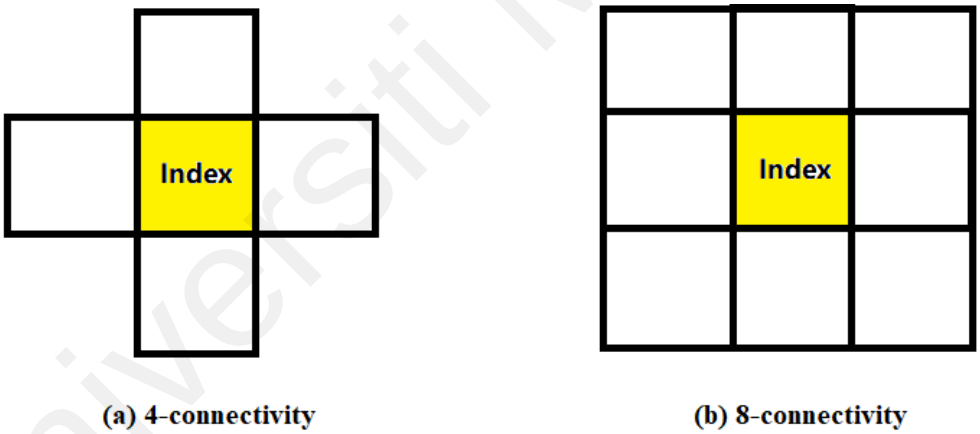


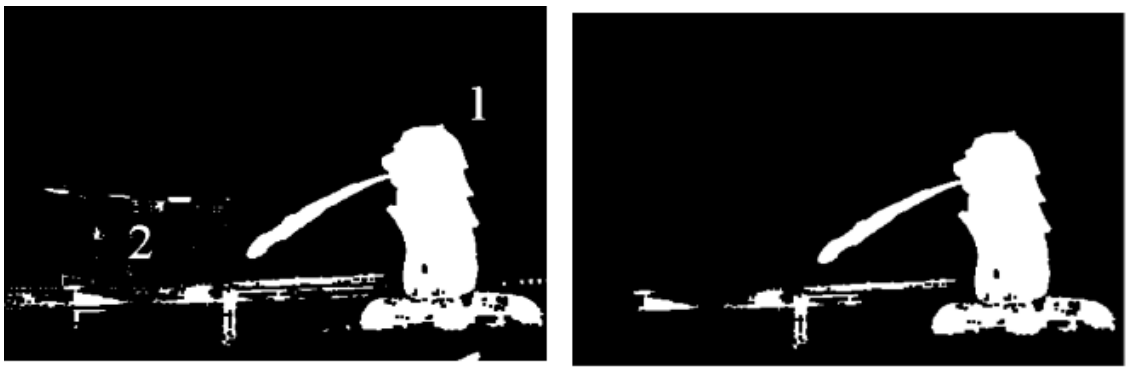
Figure 3.14: Connected components

Using an 8-connectivity makes the algorithm more robust since more pixels are included in a connected domain. However, this setting also has drawbacks. If numerous connected domains are close to others (only a 1-pixel gap), two adjacent components may be considered one. This problem can barely happen in real life since there is usually a clear gap between two nearby buildings. As long as the image is not taken from an extremely long distance and leave at least two or more pixels distance between objects, such a problem will have a slight chance to happen.

The proposed algorithm uses the OpenCV implemented method cv2.connectedComponentsWithStats which labels 8-connected domain using block-based connected components labelling method (BBDT) to do image segmentation to reduce the mistakes caused by discontinuity (Grana et al., 2010).

The proposed algorithm uses the 8-connectivity in the image segmentation to reduce the mistakes caused by discontinuity. It is observed that tiny objects may not appear as visually attractive but, for some reason, have some pixels with high-intensity value. It is also worth noting that tiny objects in the image are not even objects in real life most of the time. However, those objects cannot be denoised by filtering or morphology operation because they are more significant than the noise pixels but significantly smaller than landmark candidates. These objects sometimes mislead the algorithm at the foreground and background separation, falsely identifying them as landmark candidates. The proposed algorithm assumes that a qualified landmark should be significant enough (at least above the average size among all objects find on an image).

Figure 3.15 shows an image before and after segmentation. Figure 3.16 shows the codes to perform the segmentation. The algorithm begins by assuming all connected domains as valid candidates. Then, the algorithm filters connected domains bigger than the average size of all objects as landmark candidates. What remains after the segmentation is, hypothetically speaking, an image containing only the landmark candidates. Lesser objects to process are expected to accelerate the computational speed. The segmentation method completes the pre-processing module for this dissertation.



Original segmentation result Segmentation with area restriction

Figure 3.15: The image before and after the segmentation

```
91      retval, labels, status, centroids = cv2.connectedComponentsWithStats(t,8,cv2.CV_32S)
92      #Average area of all connect areas, only the one bigger than average will be consider
93      areaSum=0
94      for i in range(retval):
95          areaSum=areaSum+status[i][4]
96      #minus background pixels
97      aveAreaSum = (areaSum-status[0][4])/(retval-1)
98      #uint8 is image container
99      objectLabel = 1
100     objectImage = labels.copy()
101     for i in range(objectImage.shape[0]):
102         for j in range(objectImage.shape[1]):
103             # if (objectImage[i][j] != objectLabel):
104             #   objectImage[i][j] = 0
105             if (status[labels[i][j]][4]<aveAreaSum):
106                 objectImage[i][j] = 0
```

Figure 3.16: Codes for image segmentation

3.5 Module 2: an algorithm for visual salience indicators

The previous section describes the pre-processing module, which outputs an image with a group of connected domains representing landmark candidates. In this section, the salience calculation is performed on the landmark candidates. Three indicators make up the salience calculation on a landmark's visual appearance; size, colour, and shape.

3.5.1 Size salience

The size salience can be computed by counting the pixels that constitute the object. In the algorithm, size salience is computed using:

$$S_f = \sum_{i=1,j=1}^{m,n} x_i \, y_j \quad \text{.....(7)}$$

S_f from Equation (7) represents the pixel collection of a connected domain, and, m and n stand for the largest x and y coordinate values, respectively. Each connected domain is considered a potential landmark. Counting the pixels of a connected domain is sufficient to describe that landmark's size or surface area, and the method is straightforward. Figure 3.17 shows the codes that calculate landmark candidate's size.

```
148     #array big_areas_cor stores the pixel sets for each potential landmarks
149     big_areas_cor = []
150     for areas in big_areas:
151         temp = []
152         for row in range(labels.shape[0]):
153             for col in range(labels.shape[1]):
154                 if(areas[0]==labels[row][col]):
155                     temp.append([row,col])
156         big_areas_cor.append([areas[0],temp])
```

Figure 3.17: Codes for size salience

3.5.2 Colour Salience

Colour salience for a landmark candidate can be computed following Equation (2) introduced in Section 2.3.3.5. Equation (2) by Peter et al. (2010) is repeated here to ease reading:

$$S_c = 75\% * H + 20\% * S + 5\% * V \quad \text{..... (2)}$$

Where H, S and V refers to hue, saturation and value, respectively. Compared with the RGB colour space, the HSV colour model is robust to brightness changes and closer to human colour cognition. Objects with bigger S_c is considered more attractive colour-wise. According to Labrecque et al. (2012), blue is usually more visually apparent than red and can mean the colour is more saturated (pure) or the object is lighter. The module runs the codes in Figure 3.18 on each landmark candidate. The code first converts each pixel of the landmark candidate to HSV. The colour salience for each landmark candidate is calculated following the weights of H, S and V in Equation (2).

```
171     #ele structurer [label,[[x,y],[x1,y1].....[xn,yn]]
172     for ele in big_areas_cor:
173         color_temp = 0
174         for coordinate in ele[1]:
175             #coordinates test, all big_areas are marked black
176             # if(ele[0]==47):
177             #    img_test[coordinate[0]][coordinate[1]]=[0,0,0]
178             #Hue, Saturation, Value
179             color_temp=color_temp+hsv[coordinate[0]][coordinate[1]][0]*0.75
180             +hsv[coordinate[0]][coordinate[1]][1]*0.20+hsv[coordinate[0]][coordinate[1]][2]*0.05
181         #caculate the avearage value and the order is: label,color salience, area
182         color_ave.append([ele[0],color_temp/status[ele[0]][4],status[ele[0]][4]])
183
```

Figure 3.18: Codes for colour salience

3.5.3 Shape Salience

A standard curvature threshold can filter the landmark candidates in getting their shape's curves and contours scores. The assumption here is that the more dynamics a shape has in curves and contours, the more unique and distinct the shape would look. Section 2.3.3.6 introduces Equation (3), a curve and contour function, to compute the landmark candidate's shape. The equation selects three adjacent neighbours and assumes these three points are on the same quadratic function. After that, the unknown constants for the quadratic function can be figured out. Since the shape for a quadratic function is continuous, the quadratic function is appropriate for curvature and contour calculation.

Feature value is calculated using the pixels that consist of the contour of an object. Equation below considers every three consequent points in a counter-clockwise order on the same quadratic function. Although using such a function to fit three consequent pixels is not accurate, it estimates the curve change of these three pixels. In real life, theoretically, there are infinite points of an object. However, due to the limitation of computers, it is only possible to use an appropriate approach to represent them. A quadratic function can well represent the local curves at the edges because the edge of an object should be smooth and continuous. Equation (3) by Roser et al. (2012) is repeated here for ease of reading:

$$C = \frac{x''y' - x'y''}{((x')^2 + (y')^2)^{\frac{3}{2}}} \quad \text{.....(3)}$$

Say three consequent geometrical pixels $(xn-2, yn-2)$, $(xn-1, yn-1)$ and (xn, yn) is following the clockwise order of pixels in the same neighbourhood. A point, for example, $(xn-2, yn-2)$ and $(xn-1, yn-1)$, is the nearest neighbour to $(xn-2, yn-2)$. For $(xn-1, yn-1)$ and (xn, yn), the nearest neighbour is $(xn-1, yn-1)$. It means that two pixels belonging to the same object will always have a third pixel to make up the neighbourhood system.

Consider three consequent points $(x1, y1), (x2, y2)$ and $(x3, y3)$ working as a small group. The middle point, $(x2, y2)$, is the value where shape saliency is calculated for each group after the quadratic function is figured out. Then the function C of Equation (3) calculates the curve and contour value representing the shape salience at point $(x2, y2)$. The process is repeated for all neighbouring groups for each landmark candidate.

```
265     for contours in conCor:
266         flag=True
267         for salienceArray in color_ave:
268             #curvature array
269             ka = []
270             #direction
271             no = []
272             #coordinates
273             po = []
274             if(contours[0]==salienceArray[0] and flag):
275                 my_arr=contours[2]
276                 # if(contours[0]==210):
277                 #   print(my_arr)
278                 po,no,ka=curvatureArray(my_arr)
279                 contour_salience = sum(abs(x) for x in ka)
280                 salienceArray.append(contour_salience)
281                 flag=False;
```

Figure 3.19: Codes for shape salience

3.5.4 Final landmark selection

This section presents the steps in deciding the final landmark among all landmark candidates. The quantifier gives individual visual salience scores for each landmark to assess the performance of the indicators separately. However, which landmark has the

overall best saliency? Which one is the most visually significant to most humans? To answer this, one needs to assess the performance of the indicators as a combination solution. However, the individual salience calculator cannot be parameterized into the same equation for the combination without normalization. Therefore, normalization is considered.

The individual salience calculator outputs an array with three-set elements representing size, colour and shapes salient for all landmark candidates. The values are kept as real numbers. Parameterizing the three type of salience value (size salience, colour salience and shape salience) as a combination solution without normalization introduces bias since the size salient raw value is more significant than the colour and shape salience. Assuming i refers to a specific landmark for m potential landmark candidates, the size salience normalization can be derived as follows:

$$S_{size} = \frac{N_i}{\sum N_{total} = N_1 + N_2 + \cdots + N_m} \quad \text{..... (8)}$$

Where N_i refers to number of pixel for object i. The idea here is to calculate the percentage of object i when considering all potential landmark pixels of the image.

Colour salience is calculated by the average HSV value of an object. The reason is that inside one image, objects commonly have different sizes. If the colour salience is determined by the sum of each pixel value of an object i, it is not fair for those small objects with bright colours. Sometimes, small objects can never match the bigger ones in terms of colour salience due to the significant difference in pixel numbers. Thus, normalizing the colour salience is proposed following:

$$S_{colour} = \frac{Colour_i}{\sum Colour_{total} = Colour_1 + Colour_2 + \cdots + Colour_m} \quad \text{..... (9)}$$

The same consideration is given to shape salience, where $Colour_i$ refers to the average HSV score for object i. $Shape_i$ refers to the curvature line score for object i. The normalizing equation for shape salience object i for all landmark candidates m is given by:

$$S_{shape} = \frac{Shape_i * W_{special}}{\sum Shape_{total} = Shape_1 * W_{speicial} + Shape_2 * W_{speicial} + \cdots Shape_m * W_{speicial}} \quad \text{..... (10)}$$

A special weight, $W_{special}$, is assigned to balance the calculation of total shape salience as large objects have much more pixels than smaller ones, resulting in larger shape salience values by default. Small objects will take a tiny proportion of normalized shape salience without balancing. The weight is given by:

$$W_{special} = 1 - N/N_t \quad \text{..... (11)}$$

Where N refers to the number of pixels of the object and N_t refers to the total pixels of potential candidates. If an object has more pixels, the corresponding shape salience is smaller based on the proportion. The more pixels an object has, the smaller the proportion will be. The normalization offsets the saliency calculation while reducing biases.

Once the salience indicators are normalized, they can be parameterized into the same equation for combination. Evaluating individual salient indicators as a combination solution requires special weights assigned to each indicator. Here, different weights are proposed to test each indicator's influence on the combination. This study proposes 45%

(W_{size}), 35% (W_{colour}) and 20% (W_{shape}) weight assignment to size, colour, and shape following:

$$S_{salience} = S_{size}W_{size} + S_{colour}W_{colour} + S_{shape}W_{shape} \quad \text{..... (12)}$$

$$\text{Where } 1 = W_{size} + W_{colour} + W_{shape} \quad \text{..... (13)}$$

The assumption is that size has the most significant influence in extracting urban landmarks followed by a colour and interesting shape (Hussain et al., 2018; Caduff & Timpf, 2008).

3.5.5 Marking landmark contours

There are several outcomes for any given landmark candidate; the largest among the candidates, the brightest among the candidates, or the most uniquely shaped. Considering the combination salience solution, any given landmark candidate can have the highest score for all three salient visual indicators. A landmark candidate can also fail the salient visual indicators, scoring the lowest individually or as a combination solution. This section proposes bounding lines to mark the contour of landmarks.

This dissertation considers labelling landmarks according to their salience outcome to preserve information. The labelling uses different colours to represent different saliency calculations. For example, the grey colour for the biggest size, green for the brightest colour, and yellow for the unique shape. Black is selected to mark the final landmark candidate, which scores the highest on the combination solution. Others that did not cut the marks are left as blue. Figure 3.20 shows the codes that assign colours according to the salience calculation.

```
376     #Biggest area with grey color
377     color[biggest_index]=[155,155,155]
378     #Green
379     color[maxColor] = [0,255,0]
380     #YELLOW
381     color[maxShape] = [0,255,255]
382     #BLACK
383     color[idxLandmark]=[0,0,0]
384     img_color = im.copy()
385     for rows in range(labels.shape[0]):
386         for cols in range(labels.shape[1]):
387             img_color[rows][cols]=color[labels[rows][cols]]
```

Figure 3.20: Codes for landmark marking

3.5.6 Algorithm output design

The output of the salience calculation algorithm has two parts: image output and salience output. The image output produces six images total entitled: “origin”, “GaussianBlur”, “threshold”, “connected domain”, “contour_my” and “coloured” while salience output produces information in the form of an array with elements such as ‘ID’, ‘colour salience’, ‘size salience and ‘shape salience’.

For image output, “origin” refers to the original input image. “GaussianBlur”, as the name suggests, is the output after an image is filtered by Gaussian filter. The resulting image for this phase will be different following users’ decisions on the filtering. An enum class called “filterMode” is provided. The “filterMode” has two components. One is a traditional Gaussian filter with a sigma value of 1.5. The other is a bilateral filter option with sigma colour 100 and sigma space 5. “Threshold” is similar to “GaussianBlur” with a customized thresholding method and Otsu’s method. The “connected domain” and “threshold” display the output images after corresponding processing.

The salience output is more straightforward than the image output. After all the processes, the algorithm finally returns an array containing all salience information, including size, colour and shape. Table 3.2 shows the 2D array structure with four elements to each object ID in each row.

Table 3.2: Saliency result in 2D array metric

$Object_1$	S_{colour_1}	S_{size_1}	S_{shape_1}
$Object_2$	S_{colour_2}	S_{size_2}	S_{shape_2}
$Object_n$	S_{colour_n}	S_{size_n}	S_{shape_n}

3.6 Algorithm testing

Another dataset is used to evaluate the performance of the landmark selection algorithm. The data source contains new aerial images taken by a parrot bebop drone or a quadcopter with a bird's eye view from 20-125 feet above the ground. The parrot bebop has an A-14 megapixels fisheye camera that enables the drone to capture images with a frontal feed. Consideration of the dissertation's scopes is given, i.e., (1) require a distinct spatial gap between landmark candidates and (2) all landmark candidates can be seen without weather obstruction like fogs or hidden behind landmark candidates or other objects.

A survey is conducted to see the human perception using the same drone dataset. The survey listed the image with the salience indicators checklist. The human is asked to decide on a salient urban landmark by using a marker pen to outline the landmark contours. Then, the human is asked to tick which indicators on the checklist influence their decision making. The human survey results are compared against the proposed model's selection. The critical aspect in the evaluation is whether the model matches the human's selection consistently.

3.7 Summary

This chapter describes the proposed algorithm's two modules that address the objectives of this dissertation. Module 1 addresses Objective 1, which segments urban landmarks through bilateral filtering, foreground and background thresholding and morphology operation. Module 2 addresses Objective 2, which is to quantify visual salience indicators of the urban landmarks through pixel counting for size, HSV brightness extraction, and curve and contour calculators. A combination solution is proposed at the end of Module 2 to evaluate urban landmark visual saliency. Model testing methodology is also described. The following chapter showcases the proposed algorithm performance compared to a human’s perception and selection of the urban landmarks.

CHAPTER 4: RESULTS AND DISCUSSION

4.1 Overview

This chapter provides the results of the proposed algorithm analysing the performances of Module 1 and Module 2. This chapter also addresses Objective 3, comparing the performance of the proposed algorithm in extracting urban landmarks between the robot and humans.

4.2 Module 1: result and discussion on landmark segmentation

In this section, five images taken using a drone flying over UM campus are used. The drone captures aerial views of several landmarks surrounding the UM campus during the daytime. Only one of the images is a frontal view and taken at nighttime. There was no way to fly the drone outside of UM campus due to the movement control disorder in Malaysia during the Covid-19 pandemic. However, the UM campus buildings are plenty and sufficient to complement the dataset. Figure 4.1 shows the raw images labelled as a_0, b_0, c_0, d_0, and e_0.

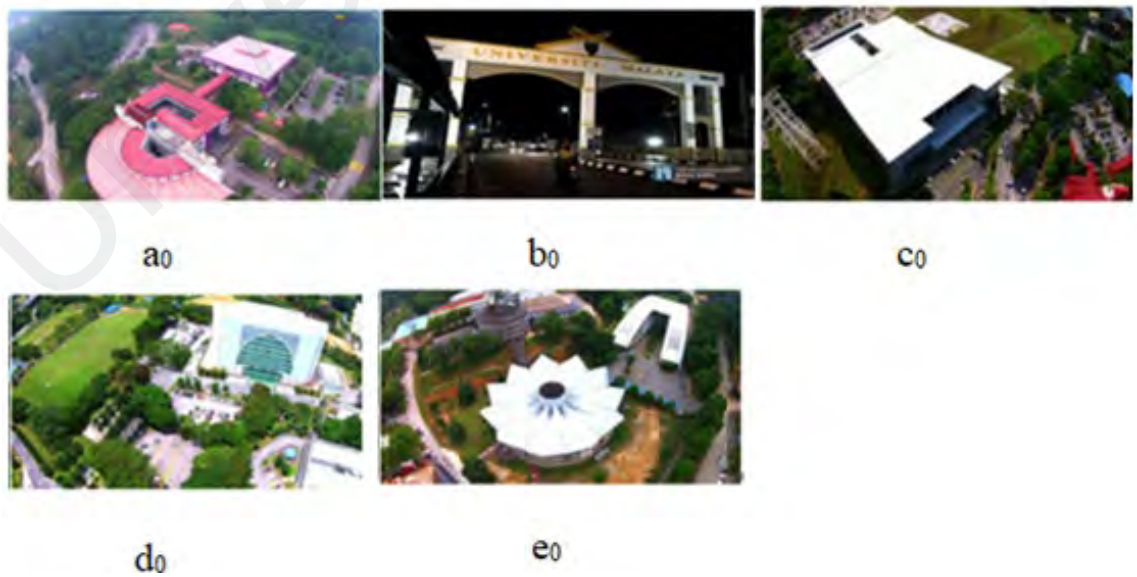
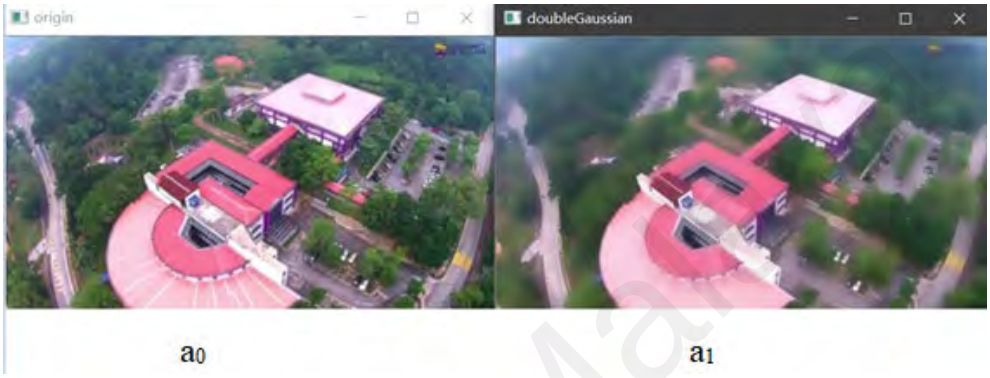


Figure 4.1: Testing images captured using a drone at the UM campus

4.2.1 Gaussian and bilateral filtering results

Figure 4.2 to Figure 4.6 are the filtering results for the original images from Figure 4.1. The filtered results show the edges information is kept (although hard to observe) while slight noise and other unwanted scalar pixels are removed from the original images. The filtered images are generally smoother, too, compared to the original images.



a_0 a_1

b_0 b_1

c_0 c_1

Figure 4.2: Original images (a_0, b_0, c_0, d_0, and e_0) and the filtered images (a_1, b_1, c_1, d_1, and e_1)

4.2.2 Foreground and background separation results

The drone used in the experiment has a limited communication range; thus not advisable to fly beyond 125 feet above the ground. The UM campus landscape separates buildings, so there are usually two or three buildings in each image at that flying height. These buildings are also the centre of attention for the images, with the histogram showing apparent peaks separating the foreground and background. Therefore, when running the five UM images' foreground and background separation codes, they pass Otsu's thresholding. Otsu's method proofs to give good performance with bimodal images. Figure 4.3 shows the thresholding results.

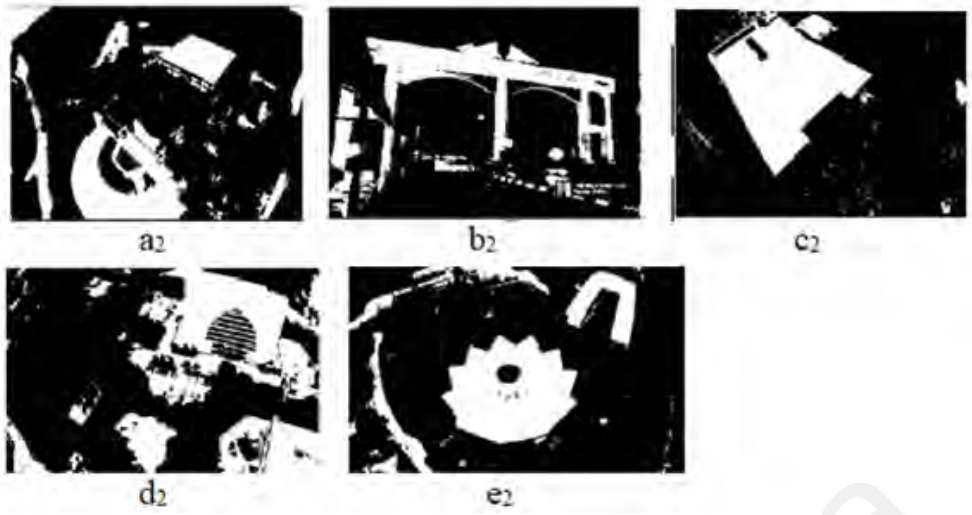


Figure 4.3: The thresholding results a_2, b_2, c_2, d_2 and e_2. The images appear bimodal, favouring and passing Otsu's method of thresholding.

4.2.3 Morphology operation results

The morphology operation closes any disconnected parts of the same building. Figure 4.4 shows the five UM images after the morphology operation. Overall, all five images show the landmarks are preserved as fully connected domains in each image. The erosion and dilation processes of the morphology operation also take care of tiny objects and object noises. Notice the centre part of e_3 in Figure 4.4, which is now clean of the black dots at the foreground and background separation stage (see e_2 in Figure 4.3 for comparison).

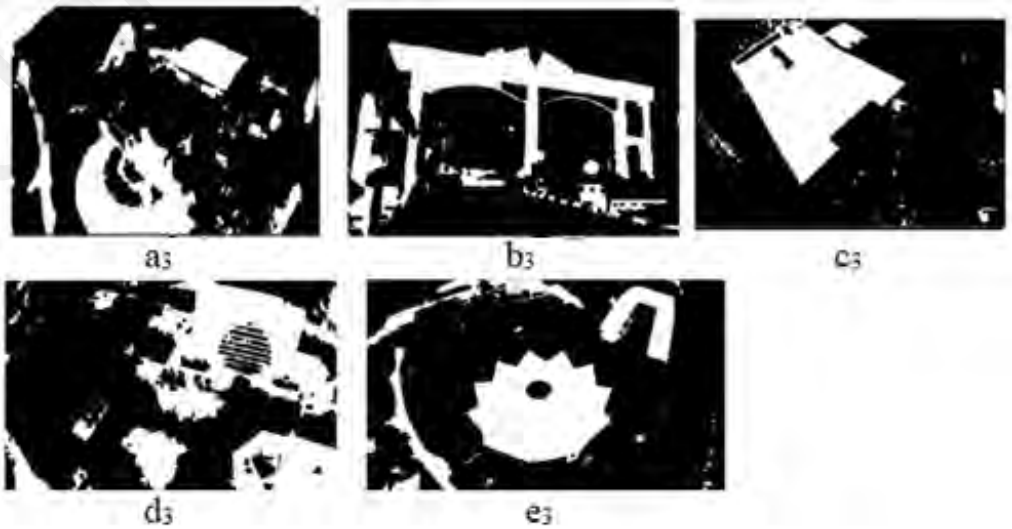


Figure 4.4: Morphology operation results in a_3, b_3, c_3, d_3 and e_3.

4.2.4 Image segmentation results

The segmentation follows the morphology operation by enforcing the removal of tiny objects and noise smaller than the average size of the image. Figure 4.5 shows the segmentation results. The segmentation process removes any tiny objects outside the connected domains, leaving only landmark candidates in the image.

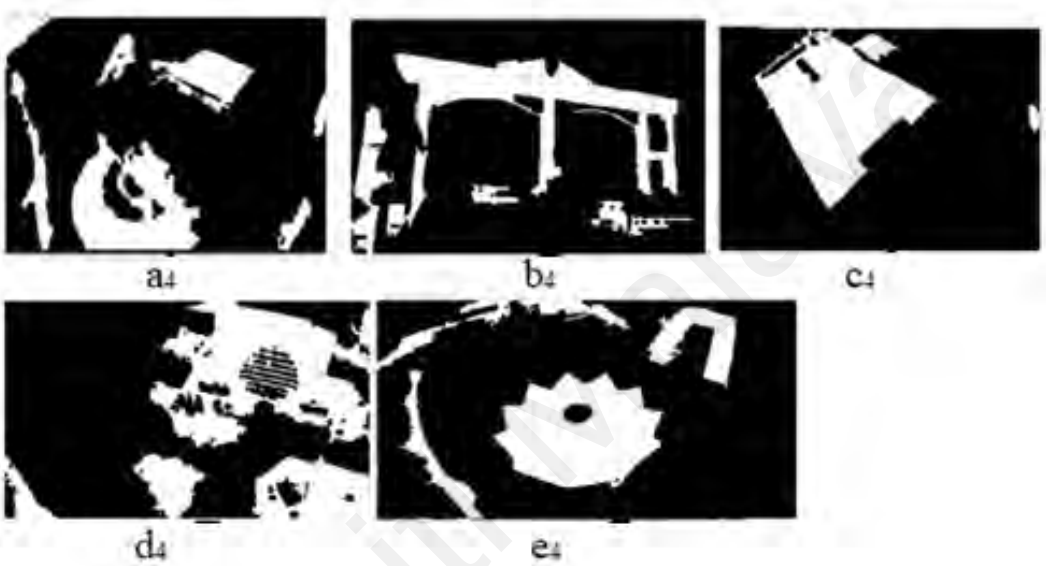


Figure 4.5: Segmentation results a_4, b_4, c_4, d_4 and e_4.

4.3 Module 1: performance as a landmark segmentation tool

A dataset containing 55 images is proposed in this section to evaluate the performance of the Module 1 algorithm as a general tool in urban landmark extraction. The images come from two sources, nine from a drone flying over the UM campus and 44 online images of well-known landmarks worldwide. The images are a mixture of aerial and frontal views of the landmark. The dataset includes a self-drawn geometric image and an urban landscape blueprint for various purposes. Figure 4.7 shows the original images for the dataset.

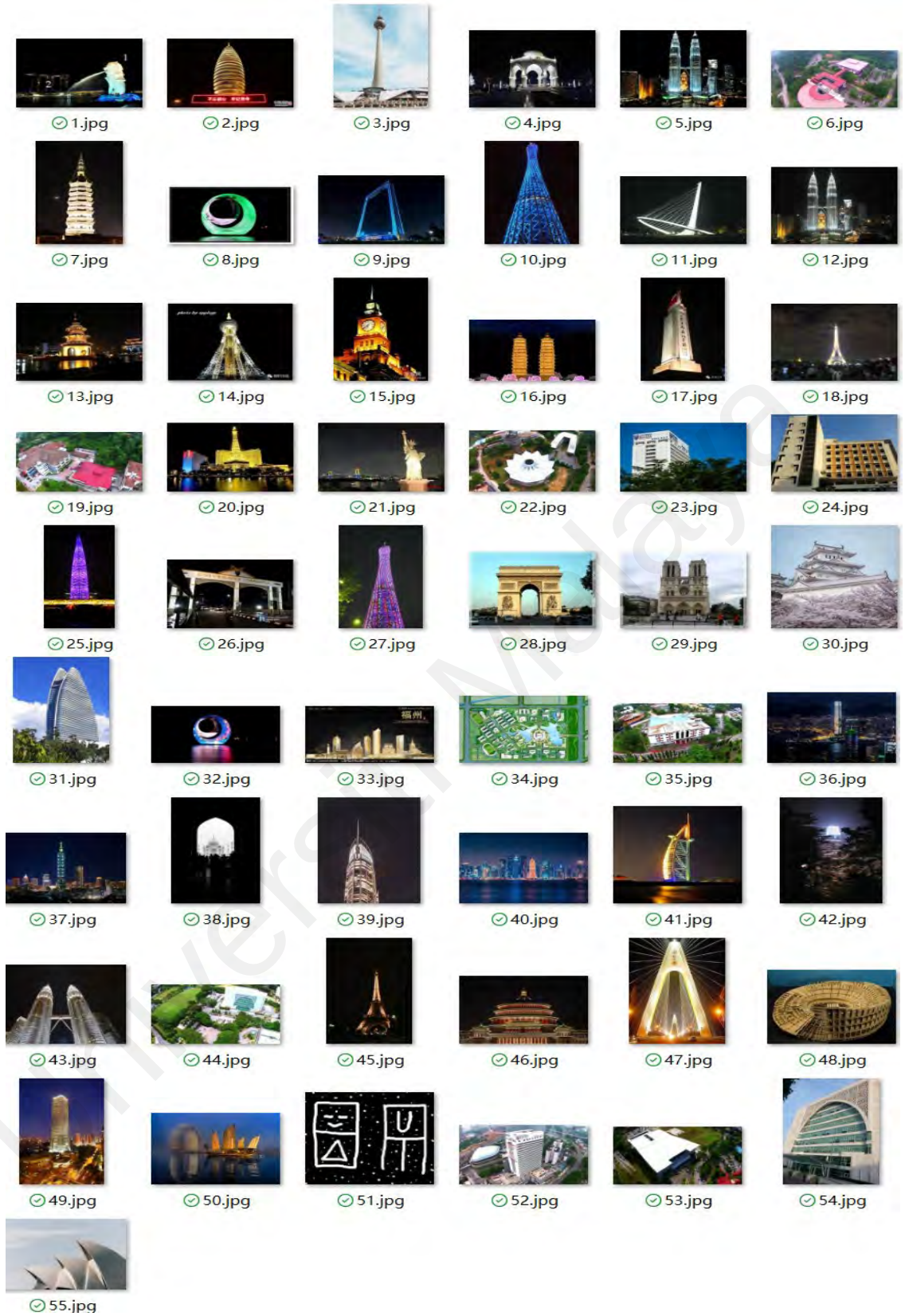


Figure 4.6: Original images of landmarks taken using a drone and from online resources. One image is self-drawn and included for variety.

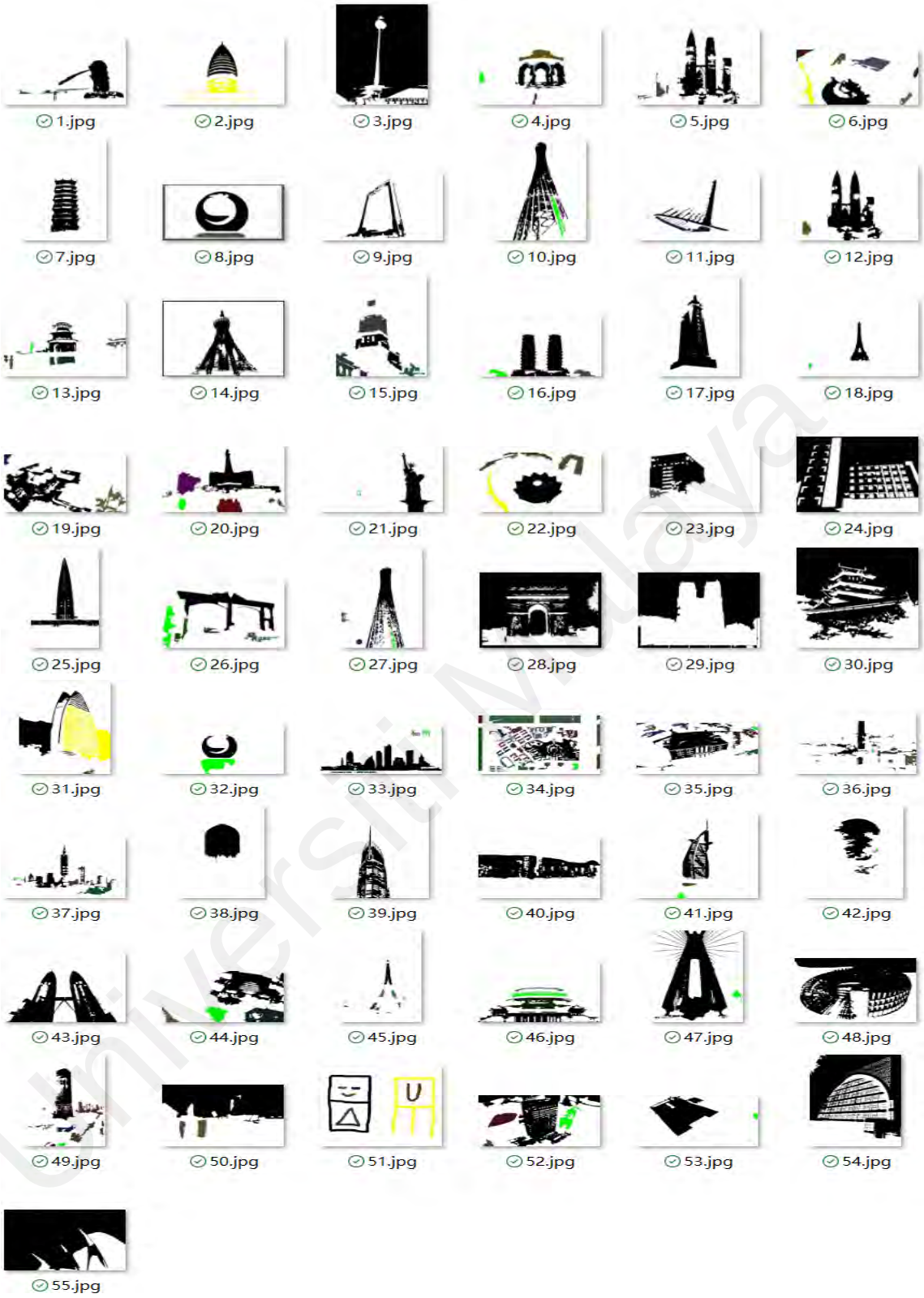


Figure 4.7: Results for the 55-image dataset (yellow and green are coloured after the whole process to mark the part that has high shape and colour salience).

Figure 4.6 shows the 55 images of the dataset, while Figure 4.7 has the segmentation results. The results show that some images are correctly extracted objects that are landmark candidates, but some fail due to issues with intensity value. Figure 4.9 shows Module 1 performance on the 55 image dataset. It is encouraging to note that the complete landmark extraction achieves 87% performance. When the background has a higher intensity value over the foreground, the discontinuity increases so much that the morphology operation cannot connect the separated parts. The connectedness issue misled the algorithm to either extract only parts of the correct landmarks or missed the landmark altogether. Figure 4.10 shows the algorithm missing extracting the correct landmark candidates in four images. Figure 4.11 shows the outcome of the partial landmark extractions in three images.

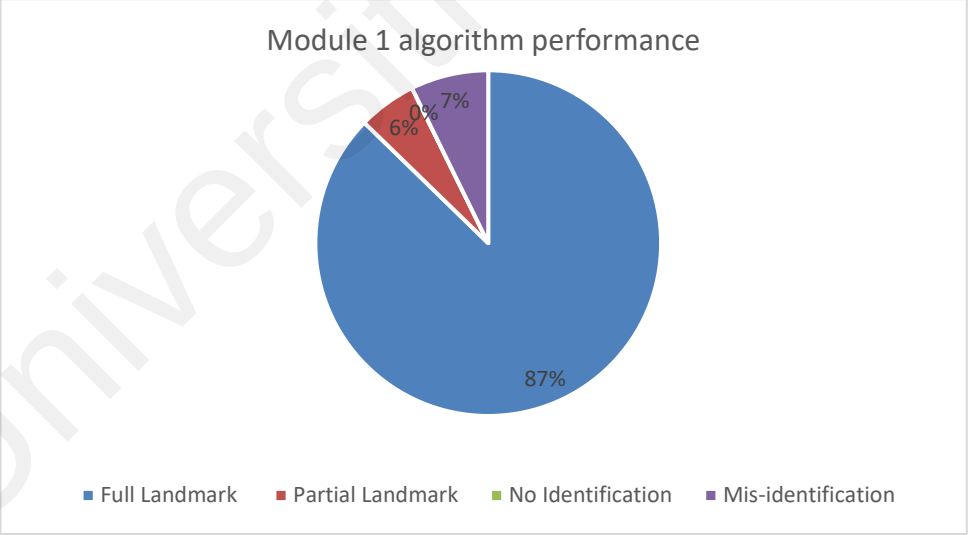


Figure 4.8: Module 1 algorithm performance

Figure 4.9: The algorithm missed segmenting the correct landmark candidates in four out of 55 images

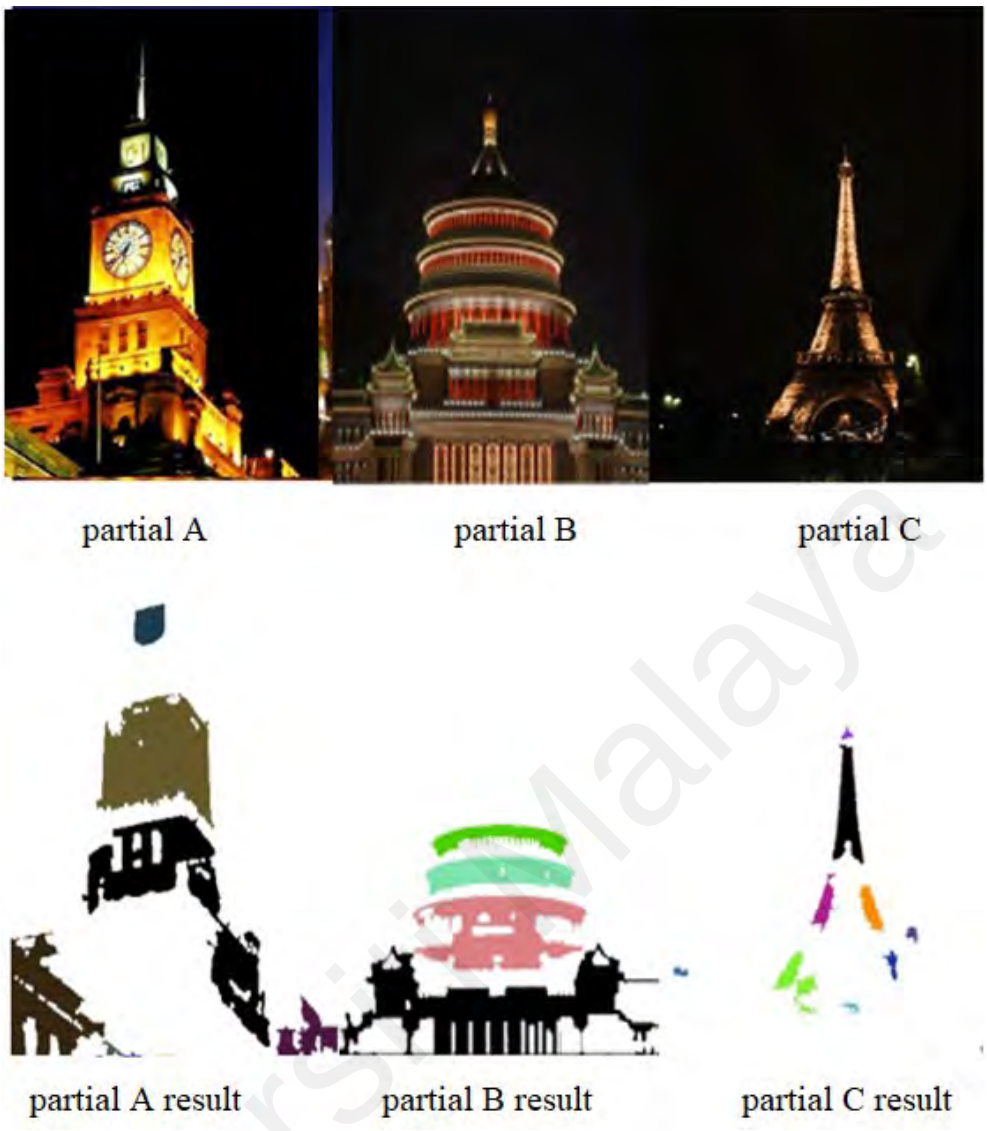


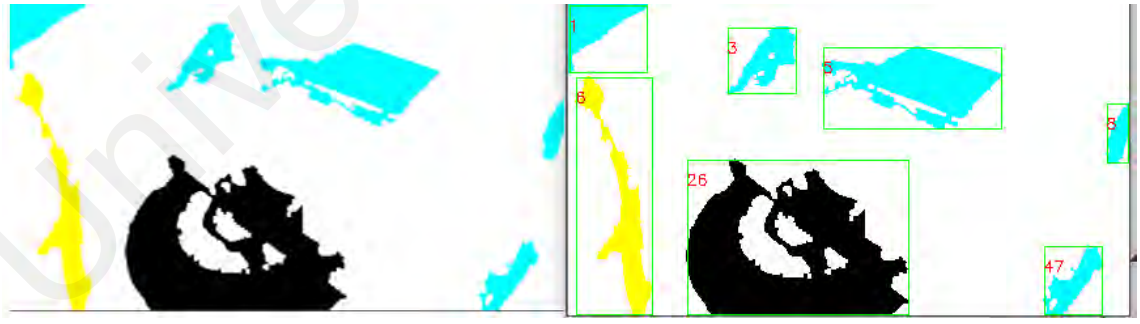
Figure 4.10: The algorithm extract only parts of the correct landmark candidates in three out of 55 images

4.4 Module 2: result and discussion on visual salience indicators

This section presents the visual salience indicators results. Figures 4.11 to 4.15 depict the saliency evaluation for images a_0, b_0, c_0, d_0 and e_0, respectively. Each image features the landmark candidates according to the saliency outcome. The colour green denotes the landmark candidate with the highest colour salience results, while yellow denotes the highest score for shape salience. The colour black represents the final candidate that scores highest in size and at least one other salience indicator. Cyan denotes the candidate who cannot top any individual salience indicators.

The results also feature the 2D array with elements representing the score for each salience indicator; colour, size and shape. The score is normalized for each indicator, and the results are rounded to 3 decimal places. Theoretically, each image result should feature four different colours, green, yellow, black and blue, rendered in this order. However, the rendering process defines if the largest-sized landmark is also the highest in either colour or shape, then green or yellow colour will be repainted to black. Therefore, there will always be one colour missing in the image.

Figure 4.11 shows the final landmark selection for original image a_0 with its saliency score. A total of seven landmark candidates are extracted in the image with IDs 1, 3, 5, 6, 8, 26, and 47. The salience score (2D array) suggests the landmark candidate ID_1 scores 0.125 for colour salience, 0.061 for size and 0.068 for shape. Meanwhile, the landmark candidate ID_3 scores 0.138 for colour, 0.05 for size and 0.13 for shape. Landmark ID_26 tops the size and colour indicators, making it the final landmark selected for the image (black). Landmark ID_6 topped the shape salience and is marked in yellow.



[[1, 0.125, 0.061, 0.068], [3, 0.138, 0.05, 0.13], [5, 0.155, 0.187, 0.206], [6, 0.145, 0.111, 0.277],

[8, 0.136, 0.021, 0.056], [26, 0.173, 0.529, 0.135], [47, 0.128, 0.041, 0.128]]↵

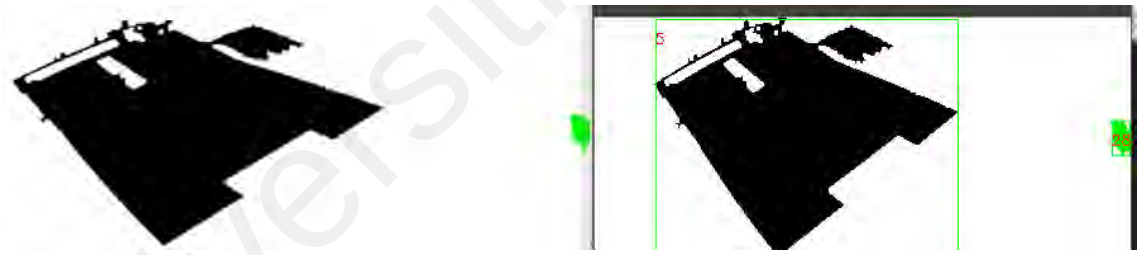
Figure 4.11: Final landmark image a_5 and the saliency score.

Figure 4.12 shows the final landmark selection for original image b_0 with its saliency score. A total of six landmark candidates are extracted in the image with IDs 2, 4, 15, 28, 35, and 42. The salience score (2D array) suggests the landmark candidate ID_2 scores

0.133 for colour salience, 0.699 for size and 0.197 for shape. Meanwhile, the landmark candidate ID_4 scores 0.218 for colour, 0.147 for size and 0.246 for shape. Landmark ID_2 tops the size and colour indicators, making it the final landmark selected for the image (black). Landmark ID_4, on the other hand, scored the highest on colour salience and is marked in green.

[[2, 0.133, 0.699, 0.197], [4, 0.218, 0.147, 0.246], [15, 0.169, 0.066, 0.196], [28, 0.11, 0.03, 0.098], [35, 0.209, 0.034, 0.105], [42, 0.162, 0.025, 0.159]]

Figure 4.12: Final landmark image b_5 and the saliency score



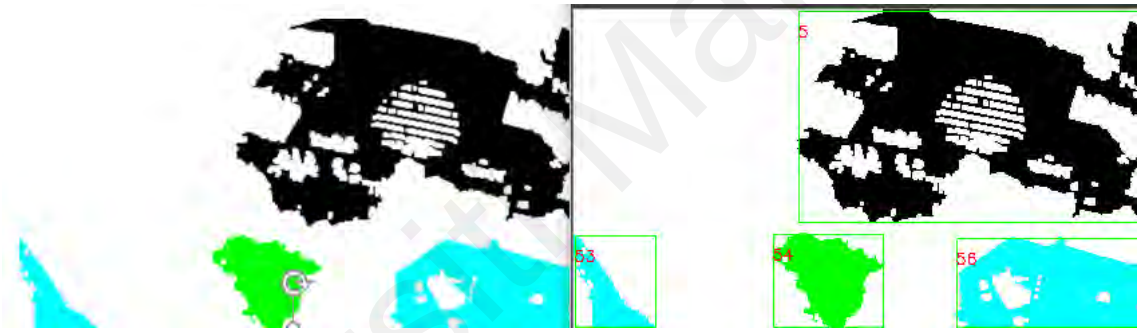
[[5, 0.32, 0.985, 0.126], [28, 0.68, 0.015, 0.874]]

Figure 4.13: Final landmark image c_5 and the saliency score.

Figure 4.13 shows the final landmark selection for original image c_0 with its saliency score. Only two landmark candidates are extracted in the image with IDs 5 and 28. The salience score (2D array) suggests the landmark candidate ID_5 scores 0.32 for colour salience, 0.985 for size and 0.126 for shape. Meanwhile, the landmark candidate ID_28 scores 0.68 for colour, 0.015 for size and 0.874 for shape. Landmark ID_5 has the biggest size salience and is the final landmark selected (black), although landmark ID_28 topped both the colour and shape salience indicators. Size trumps other indicators and given the

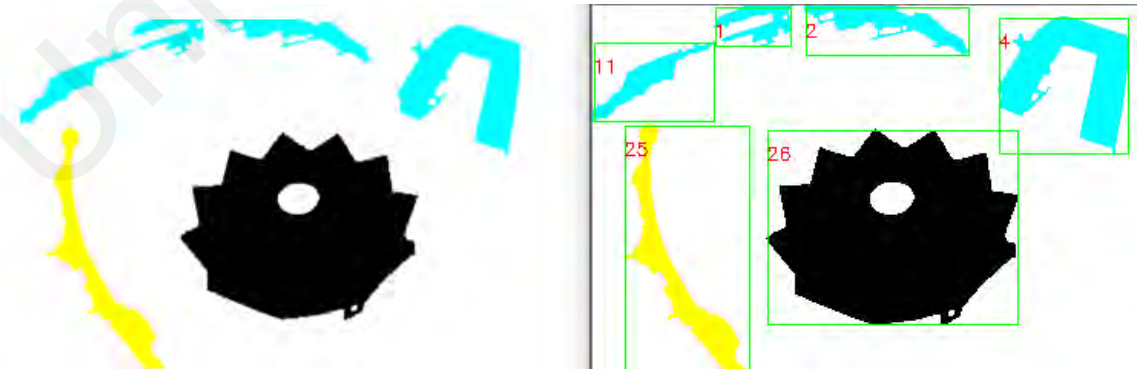
highest weight in the combination solution; 45% (W_{size}), 35% (W_{colour}) and 20% (W_{shape}). Landmark ID_6 topped the colour salience and is marked in green.

Figure 4.14 shows the final landmark selection for original image d_0 with its saliency score. A total of four landmark candidates are extracted in the image with IDs 5, 53, 54, and 56. The salience score (2D array) suggests the landmark candidate ID_5 scores 0.213 for colour salience, 0.652 for size and 0.392 for shape, making it the final landmark selected for the image (black). Meanwhile, the landmark candidate ID_54 scores 0.287, the highest for colour, and is marked in green.



[[5, 0.213, 0.652, 0.392], [53, 0.28, 0.044, 0.195], [54, 0.287, 0.098, 0.244], [56, 0.22, 0.206, 0.168]]↵

Figure 4.14: Final landmark image d_5 and the saliency score



[[1, 0.189, 0.029, 0.161], [2, 0.154, 0.055, 0.128], [4, 0.143, 0.153, 0.206], [11, 0.168, 0.038, 0.169], [25, 0.202, 0.104, 0.244], [26, 0.143, 0.621, 0.092]]↵

Figure 4.15: Final landmark image e_5 and the saliency score

Figure 4.15 shows the final landmark selection for original image e_0 with its saliency score. A total of six landmark candidates are extracted in the image with IDs 1, 2, 4, 11, 25 and 26. The salience score (2D array) suggests the landmark candidate ID_26 scores the most significant size with 0.621, making it the final landmark selected for the image (black). Landmark ID_25 topped two indicators, 0.202 for colour and 0.244 for shapes, but due to the weight distribution favouring size at 45%, landmark ID_26 is selected.

Images c_5 and e_5 show that the size salience indicator influences the final landmark selection, even though other landmark candidates occupy top spots for the colour and shape salience indicators. The normalized 2D array in Figure 4.15 may not intuitively show how significant the size indicator is for the image. The un-normalised data is presented to demonstrate size salience significance. Consider Figure 4.16, which is a replicate of Figure 4.15, except that the landmark candidates are labelled with IDs and marked with different colours for clarity.

Figure 4.16: Image e_5 with some landmark candidates repainted

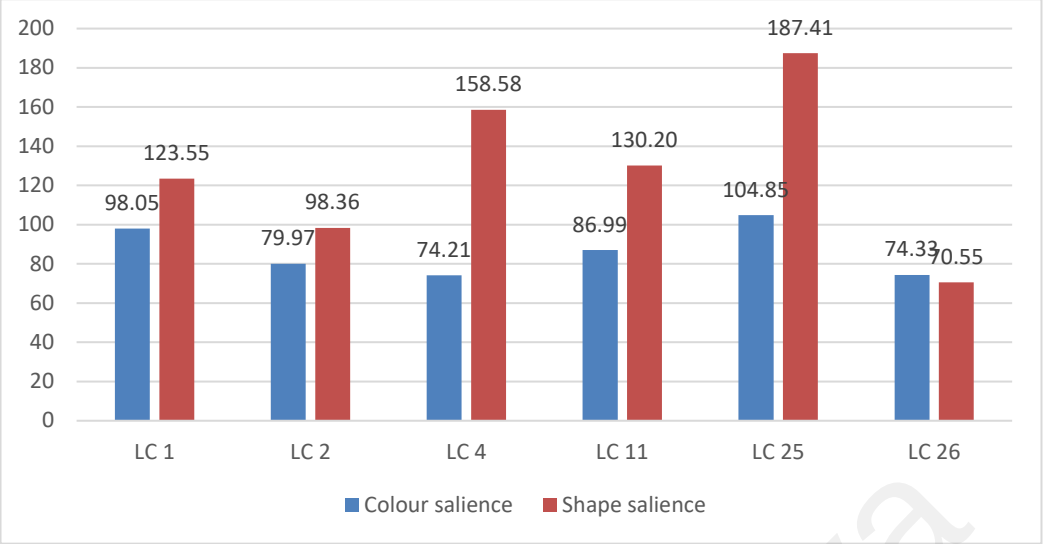


Figure 4.17: Raw data for colour and shape salience of image e_5

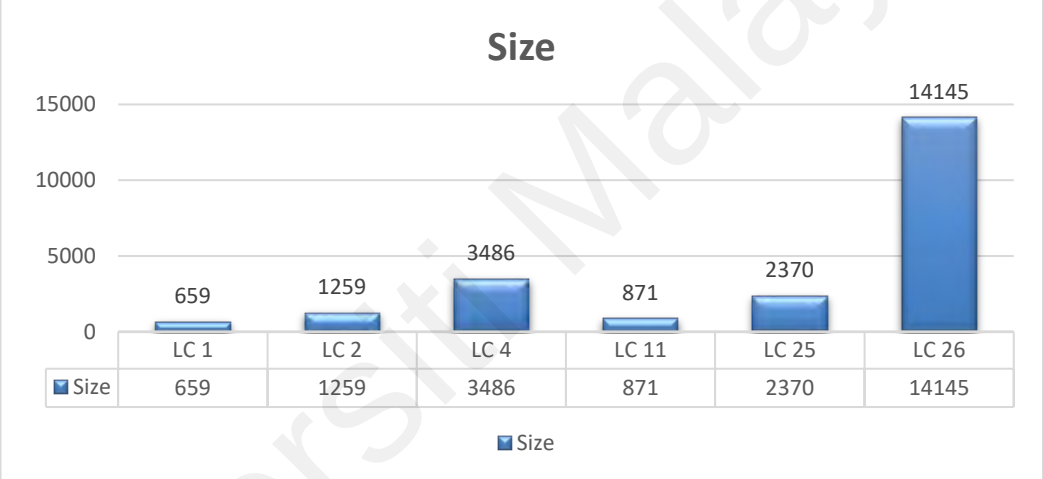


Figure 4.18: Raw data for size salience of image e_5

In Figure 4.17, no single landmark candidate (LC) scores tremendously high on the colour salience or shape salience. The LC_25 and LC_4 are the only outliers on the shape salience, and even their difference is slight. However, in Figure 4.18, landmark candidate ID_26 scores overwhelmingly high on the size salience, almost four times higher than the following closest candidates, LC_4 and LC_25. The LC_26 has to be physically prominent in the surroundings considering its context as an object to the image. The observation is in line with Hussain et al. (2018) and Caduff & Timpf (2008) proposition to increase the weight of size salience above colour and shape in a combination solution.

In this dissertation, the distribution of the weight follows 45% (W_{size}), 35% (W_{colour}) and 20% (W_{shape}) experimentation.

4.5 Module 2: performance against survey part 1

The 55-images dataset is good to test Module 1 integrity in extracting the landmark candidates. However, the assessment of landmark saliency should be done by humans, the cognitive observers of landmark features. A different human may have a different interpretation of landmark saliency. Therefore the number of images to compute matters less than the number of humans performing the landmark salience interpretation. A survey is proposed where the questionnaire includes 14 images of urban landmarks taken using a drone and random images from the internet. Sixty participants between the age of 18 and 60 participated in the survey.

Figure 4.19 shows nine images (f_0 to n_0) taken from a drone or random online images. Adding them to the initial five images; a_0, b_0, c_0, d_0, and e_0 gives the 14 images total for the survey. There were no specific criteria set for the images, and most of the online ones were selected at random. A participant spends on average 15 to 20 minutes to complete the survey, a good enough window to stay fresh and focused while completing the survey.

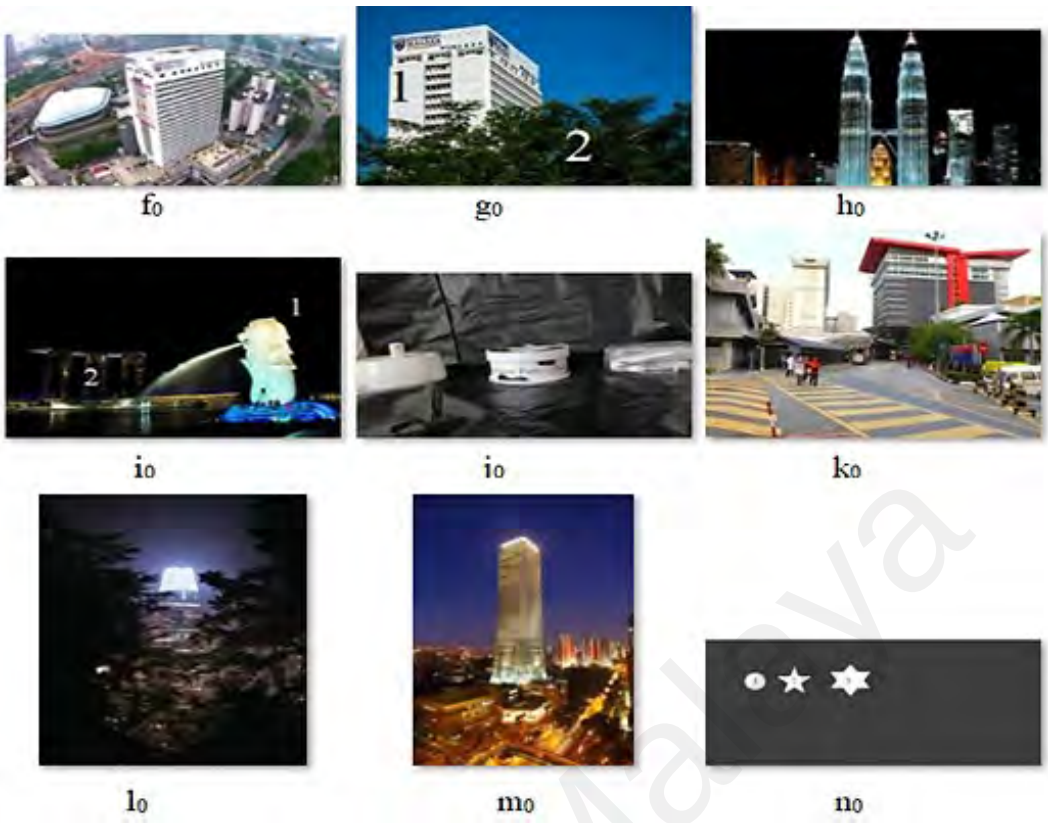


Figure 4.19: Nine other images f_0 to n_0 that makes up a 14-image testing dataset

Figure 4.20: Mismatch sample1

The results showed that 85.7% (12 out of 14) of the algorithm final landmark selection matches the 60 participants. Figure 4.20 is an example of the image that misses the participant's cut. Here, 76.7% of participants choose candidate ID_1 as the final landmark while the algorithm selects candidate ID_2. The size salience score is similar to both candidates. However, the brightness value in candidate ID_2 is superior (White color has higher intensity than red and black), hence the algorithm's selection.

Figure 4.21 shows another mismatch example. This random front view image is not showing urban landmarks but is included as variety. The results show that 46.2% of participants chose to object ID_1 while the rest selected object ID_2. No one chooses object ID_3. Participants who selected object ID_2 took a long time to choose between object ID_1 and ID_2. They eventually settle for object ID_2 because of the central position of the object. Figure 4.22 shows Module 2 match count statistics against survey results.

Figure 4.21: Mismatch sample2

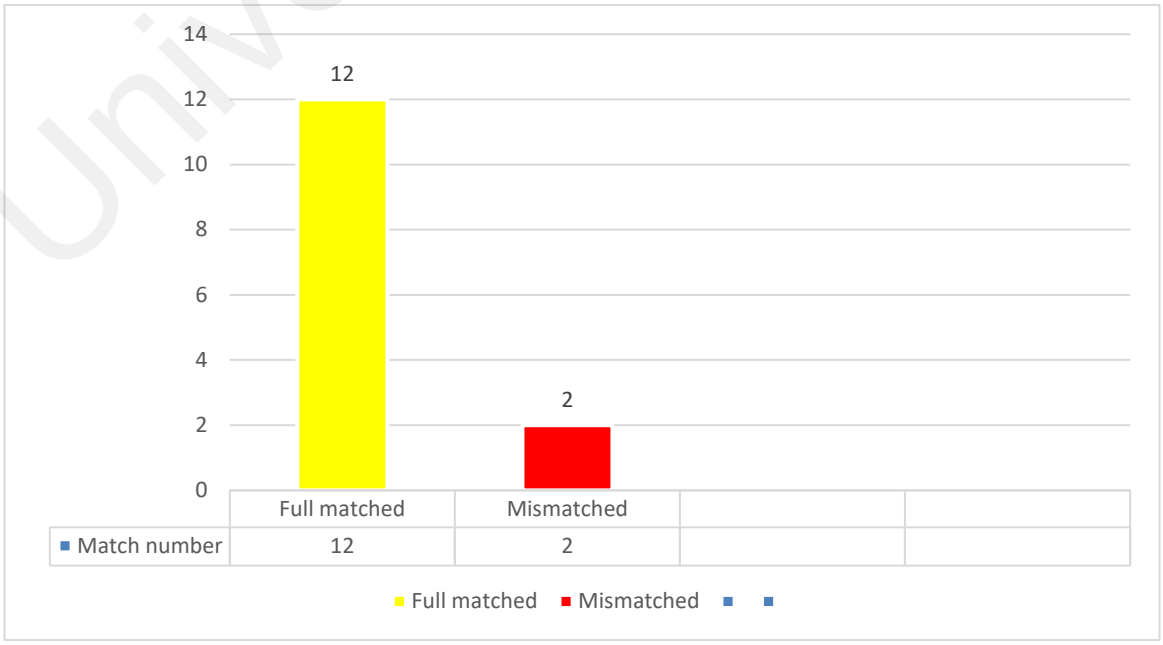


Figure 4.22: Module 2 results against survey

4.6 Module 2: performance against survey part 2

Zuhra (2016) shows a computational framework focusing on landmark saliency for robot navigation. The framework developed consists only of size and colour salience, and there is no algorithm accompanying the framework. The pre-processing, segmentation, and other image processing processes require manual intervention. Nevertheless, Zuhra (2016) surveyed human participants, and the result of the survey is interesting to this dissertation. The survey proposes images with urban landmarks, in which 19 are online images and four taken by a flying drone. The four aerial images are analyzed in this section.

Figure 4.23 shows image FA_0 selecting the bottom right-side building as the final landmark after Zuhra's framework processes it. On the contrary, her survey participants marked a different building as the final landmark in image FS_0. The algorithm proposed in this dissertation matches the participants' selection in image GA_0. This result shows that a computational solution cannot match humans' landmark preference without the shape salience indicator, even though its weight is the lowest in this dissertation.

Figure 4.24 shows the proposed algorithm does not generate satisfactory results (GA_1, GA_2 and GA_3) for images (FS_1, FS_2 and FS_3). For FS_1, the reason is that the image angle is not ideal, and the gaps between objects are not apparent, resulting in mistakes in generating connected domains. So two or more buildings are counted as one connected domain resulting in huge size saliency. Image FS_2 does not have a clear foreground and background separation threshold value. The intensity value of the parking lot, which makes up the background, is high, even higher than some buildings in the image. Therefore, the proposed algorithm considers the parking lot as a landmark candidate.

In reality, many would argue that a wide-spaced parking lot is not a bad option for a landmark. The parking lot is visually significant from the air and on land. However, parking lots are not buildings; thus are outside this dissertation scope. The proposed algorithm is also biased to size salience. It recommends the tall building as the final landmark in GA3, which Zuhra (2016) survey participants disagree with as they prefer a fancy-shaped rooftop as the final landmark (FS_3).

Figure 4.23: Framework (FA_0) and survey result (FS_0) from Zuhra (2016), and the algorithm result from this work (GA_0).

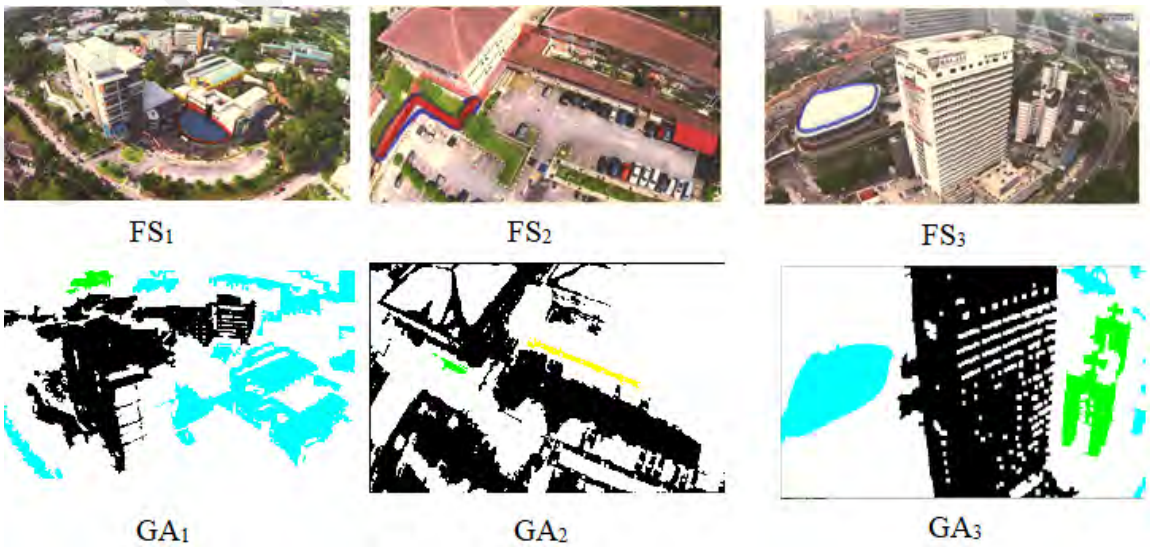


Figure 4.24: More survey results (FS_1, FS_2 and FS_3) from Zuhra (2016) that are mismatched by the algorithm result from this work (GA_1, GA_2 and GA_3).

4.7 Concluding remark

The proposed algorithm selects landmarks through three salient indicators, i.e., colour, size and shape salience. Two modules are partial to the algorithm; Module 1 focuses on landmark extraction, and Module 2 deals with saliency calculation. Module 1 performed well against 55-images dataset segmenting at 87% accuracy. Module 1 is also helpful as a general landmark extractor, following two technical scopes. The images' objects are separated and not overlapping, and background intensity is lower than foreground intensity.

Studies of human behaviour of the environment suggest that landmark saliency is second nature to humans. Humans can get attracted to objects with certain visual features; thus, humans opinions matter in evaluating Module 2. A survey with 60 participants selecting the final landmark from a set of 14 images is conducted. Module 2 algorithm achieves an 85.7% match with humans' landmark selection. Even though the number of images used is 14, getting 60 participants with varied demographics voting in sync on 12 out of 14 images is an encouraging achievement of the proposed algorithm.

However, the proposed algorithm has some limitations, evident from failed testing and survey. For example, the proposed algorithm is biased to size saliency following the 45% (W_{size}), 35% (W_{colour}) and 20% (W_{shape}) weight distribution. The weight distribution performs great when the images follow the two technical scopes. The weight distribution can amplify error whenever the segmentation has errors in generating the connected domains. Furthermore, learning from Zuhra (2016) survey work, size does not always win. There are occasions where interesting shape wins.

The proposed algorithm also does not consider criteria beyond the size, colour and shape salience indicators. When random images are used, and the landmark candidates are more or less scoring the same on the salience metric, humans often decide on the candidate's position in making a decision. The popular position to consider is the middle, which intuitively draws human attention. There are also occasions where a variety of colours become the deciding factor. The proposed algorithm does not reflect this at the point of this dissertation completion. The algorithm focuses on objects' brightness or illuminance properties through the HSV colour scale, not RGB.

4.8 Summary

This chapter presents the results and discussion on Module 1 and Module 2 performances through several experiment designs. The evaluation includes a survey with human participants to compare the algorithm proposed for the robot navigation application. The experiment and performance evaluation completes Objective 3. Both modules show encouraging performances and consistency, but at the same time, the experiments unveil several limitations for the proposed algorithm. The next chapter forwards a conclusion to the dissertation and a direction for future work.

CHAPTER 5: CONCLUSION AND FUTURE WORK

5.1 Conclusion

Urban landmarks are spatial features that are visually significant in the neighbourhood. Humans can cognitively select urban landmarks based on visual appearances like size, colour, and shapes. Many researchers have attempted to evaluate and extract visual landmarks by abstracting their features and quantifying their salience. In 2008, Caduff & Timpf presented a framework on urban landmark visual salience indicators. Their framework shows the indicators can be singular or in combination mode. Hussain et al. (2018) propose size should weigh the highest in the combination solution. Also, the indicators may or may not appear in determining the saliency factor for an urban landmark. This makes quantifying salience indicators challenging to do.

Humans use qualitative, high-level visual features of the landmark in their navigation. In contrast, robots use empirical, low-level HOG and SURF features in their landmark extraction for navigation. A quantitative model for visual salience indicators in urban landmark extraction seems beneficial to the robotics community and could improve understanding of cognitive robot navigation.

This dissertation contributed a new algorithm to computing visual landmark saliency for an application in robot navigation. The algorithm fills an important gap in the robot vision literature, i.e., developing a saliency metric for autonomous landmark selection. This dissertation proposed a combination solution for the visual salience indicators following the 45% (W_{size}), 35% (W_{colour}) and 20% (W_{shape}) weight distributions. The results are promising, achieving over 85% accuracy compared to human evaluators who participated in an evaluation survey. The development and testing complete all three objectives proposed in this dissertation.

Apart from the visual salience indicators weight distributions, the proposed algorithm relies heavily on several technical scopes to improve the salient landmark selection. The scopes are:

1. Require distinct gaps between objects in the image
2. Objects visibility is high, no heavy fogs, no obstacle/overlap with other objects
3. The foreground object should have a higher intensity value than the background object

Splitting the algorithm development into two modules is reasonable considering the integrity of the landmark segmentation directly influences the accuracy of the salient detectors. On its own, the Module 1 algorithm can be used as a general segmentation tool for use cases involving extracting large, bright or interesting-shaped objects in the image foreground. Although in the beginning, the idea is to attempt the saliency metric on aerial use cases only, the results show the proposed algorithm can handle frontal view with similar accuracy and consistency. This opens up an opportunity to extend the Module 1 use case not just on drones but mobile robots too. There has yet mobile robot works that extract meaningful visual salient landmarks in building a global map of its surrounding. Most of the robot mapping features the SURF, SIFT, corners and edges as a landmark. Hopefully, the small contribution from this dissertation can facilitate that direction.

In summary, many researchers have attempted to evaluate and extract visual landmarks by abstracting their features and quantifying their salience. This dissertation applauds the efforts and has followed suit, contributing a new saliency metric for autonomous landmark selection.

5.2 Future Work

There are several directions in the immediate future where this dissertation can go. For example, despite the performance of the Module 1 algorithm, a lot can be improved for the foreground and background separation. The issues with the histogram intensity value appearing bimodal or not affects the thresholding method used. Looking into the thresholding method is a prospect for future work.

The survey conducted to evaluate algorithm performance reveals a limitation with the HSV brightness value used. For example, humans tend to get attracted to colour contrast, but the colour contrast is not considered in this dissertation. One can add colour contrast elements in the saliency metric through the RGB colour value.

When computing shape salience, the algorithm should not automatically consider all pixels in the image. Otherwise, one risks adding noises to the shape salience indicator calculation and the results will not be accurate. Researchers who want to improve the accuracy should find a better solution for noise removal before calculating shape salience.

Size salience is also similar to colour salience. Large objects are not necessarily visually attractive at all times, so a much smaller object may also be unique to human beings. Camera angles affect an image's perspective. Big objects can look small and small objects can appear larger with crude angles. Finding a way to standardize image perspective is an interesting future work too.

The combination solution with 45% (W_{size}), 35% (W_{colour}) and 20% (W_{shape}) weights distribution does not always work. The performance of the combination solution is worth investigating through more datasets and compared with human evaluators.

The curvature value considers three adjacent geometrical pixels (the geometric distance not bigger than 1, in terms of x or y axis) on the same quadratic function. There are other ways to calculate curvature value, like using an osculating circle. Different methods may generate better results, and future researchers can try these methods for accuracy and time reduction.

Lastly, the testing should continue on the robotic platform. It would be interesting to explore salient landmarks, particularly in an indoor environment where GPS is unavailable. A robot that recognizes salient landmarks for homecoming or revisiting a place will have many domestic applications. It is also exciting to explore the performance of the proposed algorithm in recognizing consistent landmarks when multiple robots with multiple points of view are navigating in the same environment.

REFERENCES

Ahmadpoor, N., & Shahab, S. (2019). Spatial knowledge acquisition in the process of navigation: A review. Current Urban Studies, 7(1), 1-19.

Al-Amri, S. S., & Kalyankar, N. V. (2010). Image segmentation by using threshold techniques. arXiv preprint arXiv:1005.4020.

Alshammari, N., Akcay, S., & Breckon, T. P. (2018, June). On the impact of illumination-invariant image pre-transformation for contemporary automotive semantic scene understanding. In 2018 IEEE Intelligent Vehicles Symposium (IV) (pp. 1027-1032). IEEE.

Axelrod, R. (Ed.). (2015). Structure of decision: The cognitive maps of political elites. Princeton university press.

Azizul, Z., & Yeap, W. (2015). Autonomous robot mapping by landmark association. In EAP Joint Conference on Cognitive Science. CEUR-WS. org.

Bai, M. R. (2010). A new approach for border extraction using morphological methods. International Journal of Engineering Science and Technology, 2(8), 3832-3837.

Barr, J., Mizrach, B., & Mundra, K. (2015). Skyscraper height and the business cycle: separating myth from reality. Applied Economics, 47(2), 148-160.

Isa, M. D., Mohammad, A., & Hanifah, M. Z. M. (2017). Vision mobile robot system with color optical sensor. ARPN Journal of Engineering and Applied Sciences, 12(4).

Bora, D. J. (2017). Importance of image enhancement techniques in colour image segmentation: a comprehensive and comparative study. arXiv preprint arXiv:1708.05081.

Caduff, D., & Timpf, S. (2008). On the assessment of landmark salience for human navigation. *Cognitive processing*, *9*(4), 249-267.

Chandel, R., & Gupta, G. (2013). Image filtering algorithms and techniques: A review. International Journal of Advanced Research in Computer Science and Software Engineering, 3(10).

Chan, E., Baumann, O., Bellgrove, M. A., & Mattingley, J. B. (2012). From objects to landmarks: the function of visual location information in spatial navigation. Frontiers in psychology, 3, 304.

Chow, J. F., Kocer, B. B., Henawy, J., Seet, G., Li, Z., Yau, W. Y., & Pratama, M. (2019). Toward underground localization: Lidar inertial odometry enabled aerial robot navigation. arXiv preprint arXiv:1910.13085.

Chudasama, D., Patel, T., Joshi, S., & Prajapati, G. I. (2015). Image segmentation using morphological operations. International Journal of Computer Applications, 117(18).

Cui, D., Xue, J., & Zheng, N. (2015). Real-time global localization of robotic cars in lane level via lane marking detection and shape registration. IEEE Transactions on Intelligent Transportation Systems, 17(4), 1039-1050.

Do, Q. B., Beghdadi, A., & Luong, M. (2011, July). Image denoising using Bilateral filter in high dimensional patch-space. In 3rd European Workshop on Visual Information Processing (pp. 36-41). IEEE.

Dubey, R. K., Sohn, S. S., Thrash, T., Hoelscher, C., & Kapadia, M. (2019). Identifying indoor navigation landmarks using a hierarchical multi-criteria decision framework. In Motion, Interaction and Games (pp. 1-11).

Duckham, M., Winter, S., et al. (2010). Including landmarks in routing instructions. Journal of Location Based Services, 4(1), 28-52.

Epstein, R. A., Patai, E. Z., Julian, J. B., & Spiers, H. J. (2017). The cognitive map in humans: spatial navigation and beyond. Nature neuroscience, 20(11), 1504.

Fajnerová, I., Rodriguez, M., Levčík, D., Konrádová, L., Mikoláš, P., Brom, C., ... & Horáček, J. (2014). A virtual reality task based on animal research–spatial learning and memory in patients after the first episode of schizophrenia. Frontiers in behavioral neuroscience, 8, 157.

Zuhra, F. T. (2016). *A Heuristic Approach to Computing Landmark Saliency for Micro Aerial Vehicle (MAV) Navigation* (Masters dissertation, Fakulti Sains Komputer dan Teknologi Maklumat).

Gangaputra, R. (2017). Indoor landmark and indoor wayfinding: The indoor landmark identification issue. Unpublished master's thesis). Technische Universität München, München.

Götze, J., & Boye, J. (2016). Learning landmark salience models from users' route instructions. Journal of Location Based Services, 10(1), 47-63.

Goyal, M. (2011). Morphological image processing. IJCST, 2(4).

Grana, C., Borghesani, D., & Cucchiara, R. (2010). Optimized block-based connected components labeling with decision trees. IEEE Transactions on Image Processing, 19(6), 1596-1609.

Gupta, S., Fouhey, D., Levine, S., & Malik, J. (2017). Unifying map and landmark based representations for visual navigation. arXiv preprint arXiv:1712.08125.

Hentschel, M., & Wagner, B. (2010, September). Autonomous robot navigation based on openstreetmap geodata. In 13th International IEEE Conference on Intelligent Transportation Systems (pp. 1645-1650). IEEE.

Hussain, K. A. M., & Ujang, N. (2018). Identification of Landmarks in the Historic District of Banda Hilir, Melaka, Malaysia. *Asian Journal of Quality of Life*, *3*(9), 99-110.

Iida, S., & Yuta, S. (1991, November). Vehicle command system and trajectory control for autonomous mobile robots. In Proceedings IROS'91: IEEE/RSJ International Workshop on Intelligent Robots and Systems' 91 (pp. 212-217). IEEE.

Ishikoori, Y., Madokoro, H., & Sato, K. (2017, December). Semantic position recognition and visual landmark detection with invariant for human effect. In 2017 IEEE/SICE International Symposium on System Integration (SII) (pp. 657-662). IEEE.

Jeevitha, K., Iyswariya, A., RamKumar, V., Basha, S. M., & Kumar, V. P. (2020). A REVIEW ON VARIOUS SEGMENTATION TECHNIQUES IN IMAGE PROCESSSING. European Journal of Molecular & Clinical Medicine, 7(4), 1342-1348.

Kaganami, H. G., & Beiji, Z. (2009, September). Region-based segmentation versus edge detection. In 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (pp. 1217-1221). IEEE.

Kang, W. X., Yang, Q. Q., & Liang, R. P. (2009, March). The comparative research on image segmentation algorithms. In 2009 First International Workshop on Education Technology and Computer Science (Vol. 2, pp. 703-707). IEEE.

Kaur, B., & Kaur, S. P. (2013). Applications of Mathematical Morphology in Image Processing: A Review 1.

Krupic, J., Bauza, M., Burton, S., & O'Keefe, J. (2018). Local transformations of the hippocampal cognitive map. Science, 359(6380), 1143-1146.

Kunii, Y., Kovacs, G., & Hoshi, N. (2017, June). Mobile robot navigation in natural environments using robust object tracking. In 2017 IEEE 26th international symposium on industrial electronics (ISIE) (pp. 1747-1752). IEEE.

Labrecque, L. I., & Milne, G. R. (2012). Exciting red and competent blue: the importance of colour in marketing. Journal of the Academy of Marketing Science, 40(5), 711-727.

Li, Q., Zhu, J., Liu, T., Garibaldi, J., Li, Q., & Qiu, G. (2017, November). Visual landmark sequence-based indoor localization. In Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery (pp. 14-23).

Li, Y., Zhu, R., Mi, L., Cao, Y., & Yao, D. (2016). Segmentation of white blood cell from acute lymphoblastic leukemia images using dual-threshold method. Computational and mathematical methods in medicine, 2016.

Liskovec, M., & Kovarova, A. (2016, June). Beacon Based Localization Refined by Outputs from Mobile Sensors. In Proceedings of the 17th International Conference on Computer Systems and Technologies 2016 (pp. 277-284).

Loevsky, I., & Shimshoni, I. (2010). Reliable and efficient landmark-based localization for mobile robots. Robotics and Autonomous Systems, 58(5), 520-528.

Mazzeo, P. L., Giove, L., Moramarco, G. M., Spagnolo, P., & Leo, M. (2011, August). HSV and RGB colour histograms comparing for objects tracking among non overlapping FOVs, using CBTF. In 2011 8th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS) (pp. 498-503). IEEE.

Nader, J., Alqadi, Z. A., & Zahran, B. (2017). Analysis of Colour Image Filtering Methods. International Journal of Computer Applications, 174(8), 12-17.

Narkhede, H. P. (2013). Review of image segmentation techniques. International Journal of Science and Modern Engineering, 1(8), 54-61.

Nee, L. H., Mashor, M. Y., & Hassan, R. (2012). White blood cell segmentation for acute leukemia bone marrow images. Journal of Medical Imaging and Health Informatics, 2(3), 278-284.

Oskoei, M. A., & Hu, H. (2010). A survey on edge detection methods. University of Essex, UK, 33.

Pedrosa, G. V., Barcelos, C. A. Z., & Batista, M. A. (2011, May). An image retrieval system using shape salience points. In 2011 IEEE International Symposium of Circuits and Systems (ISCAS) (pp. 2797-2800). IEEE.

Peters, D., Wu, Y., & Winter, S. (2010, August). Testing landmark identification theories in virtual environments. In International Conference on Spatial Cognition (pp. 54-69). Springer, Berlin, Heidelberg.

Puthussery, A. R., Haradi, K. P., Erol, B. A., Benavidez, P., Rad, P., & Jamshidi, M. (2017, June). A deep vision landmark framework for robot navigation. In 2017 12th system of systems engineering conference (SoSE) (pp. 1-6). IEEE.

Raid, A. M., Khedr, W. M., El-Dosuky, M. A., & Aoud, M. (2014). Image restoration based on morphological operations. International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), 4(3), 9-21.

Röser, F., Krumnack, A., Hamburger, K., & Knauff, M. (2012). A four factor model of landmark salience–A new approach. In Proceedings of the 11th International Conference on Cognitive Modeling (ICCM) (pp. 82-87). Berlin: Technische Universität Berlin.

Saini, S., & Arora, K. (2014). A study analysis on the different image segmentation techniques. International Journal of Information & Computation Technology, 4(14), 1445-1452.

Senthilkumaran, N., & Rajesh, R. (2009, October). Image segmentation-a survey of soft computing approaches. In 2009 International Conference on Advances in Recent Technologies in Communication and Computing (pp. 844-846). IEEE.

Sharma, N., Mishra, M., & Shrivastava, M. (2012). Colour image segmentation techniques and issues: an approach. International Journal of Scientific & Technology Research, 1(4), 9-12.

Sharif, J. M., Miswan, M. F., Ngadi, M. A., Salam, M. S. H., & bin Abdul Jamil, M. M. (2012, February). Red blood cell segmentation using masking and watershed algorithm: A preliminary study. In 2012 International Conference on Biomedical Engineering (ICoBE) (pp. 258-262). IEEE.

Sherif, M. N. M. (2012). Istana Negara, the National Palace of Malaysia: A Symbol of Sovereignty & Majestic of the Malay Sultanate's. Maya Maju (M) Sdn Bhd with cooperation from Prime Minister's Department.

Singh, K. K., & Singh, A. (2010). A study of image segmentation algorithms for different types of images. International Journal of Computer Science Issues (IJCSI), 7(5), 414.

Smith, A. R. (1978). Colour gamut transform pairs. ACM Siggraph Computer Graphics, 12(3), 12-19.

Song, Y., Bao, L., & Yang, Q. (2014, March). Real-time video decolourization using bilateral filtering. In IEEE Winter Conference on Applications of Computer Vision (pp. 159-166). IEEE.

Sreedhar, K., & Panlal, B. (2012). Enhancement of images using morphological transformation. arXiv preprint arXiv:1203.2514.

Sutton, E. (2016). Histograms and the zone system. Illustrated Photography.

Szafir, D. A. (2017). Modeling colour difference for visualization design. IEEE transactions on visualization and computer graphics, 24(1), 392-401.

Tambe, S. B., Kulhare, D., Nirmal, M. D., & Prajapati, G. (2013). Image processing (IP) through erosion and dilation methods.

Tomari, R., Zakaria, W. N. W., Jamil, M. M. A., Nor, F. M., & Fuad, N. F. N. (2014). Computer aided system for red blood cell classification in blood smear image. Procedia Computer Science, 42, 206-213.

Tommasi, L., Chiandetti, C., Pecchia, T., Sovrano, V. A., & Vallortigara, G. (2012). From natural geometry to spatial cognition. Neuroscience & Biobehavioral Reviews, 36(2), 799-824.

Wang, L., Wang, L., & Fan, X. (2019, November). An Experiment Research on Landmark Learning underly human Spatial Cognition. In Proceedings of the International Workshop on Artificial Intelligence and Education (pp. 23-28).

Weng, M., Xiong, Q., & Kang, M. (2017). Salience indicators for landmark extraction at large spatial scales based on spatial analysis methods. ISPRS International Journal of Geo-Information, 6(3), 72.

Xi, D., Fan, Q., Yao, X. A., Jiang, W., & Duan, D. (2016). A visual salience model for wayfinding in 3D virtual urban environments. Applied Geography, 75, 176-187.

Yuheng, S., & Hao, Y. (2017). Image segmentation algorithms overview. arXiv preprint arXiv:1707.02051.

Zaitoun, N. M., & Aqel, M. J. (2015). Survey on image segmentation techniques. Procedia Computer Science, 65, 797-806.

Zhang, H., Fritts, J. E., & Goldman, S. A. (2008). Image segmentation evaluation: A survey of unsupervised methods. computer vision and image understanding, 110(2), 260-280.

Zhang, Y. (2017). Image processing. Walter de Gruyter GmbH & Co KG.

Zhao, Z., Mao, Y., Ding, Y., Ren, P., & Zheng, N. (2019, September). Visual-Based Semantic SLAM with Landmarks for Large-Scale Outdoor Environment. In 2019 2nd China Symposium on Cognitive Computing and Hybrid Intelligence (CCHI) (pp. 149-154). IEEE.

Zhu, Y., & Huang, C. (2012). An improved median filtering algorithm for image noise reduction. Physics Procedia, 25, 609-616.