

**Object Finder For The Visually Impaired**

By  
LEE JIA HUI

A REPORT  
SUBMITTED TO  
Universiti Tunku Abdul Rahman  
in partial fulfillment of the requirements  
for the degree of  
BACHELOR OF INFORMATION SYSTEM (HONS)  
INFORMATION SYSTEM ENGINEERING  
Faculty of Information and Communication Technology  
(Perak Campus)

JAN 2013

UNIVERSITI TUNKU ABDUL RAHMAN

**REPORT STATUS DECLARATION FORM**

**Title:** Object Finder For The Visually Impaired

**Academic Session:** January 2013

I \_\_\_\_\_

**(CAPITAL LETTER)**

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

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

Verified by,

\_\_\_\_\_  
(Author's signature)

\_\_\_\_\_  
(Supervisor's signature)

**Address:**

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_  
Supervisor's name

**Date:** \_\_\_\_\_

**Date:** \_\_\_\_\_

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**Object Finder for the Visually Impaired**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : \_\_\_\_\_

Name : LEE JIA HUI

Date : \_\_\_\_\_

## **Acknowledgement**

This project would not be completed without the contribution of some people. In this regard, the below people are being appreciated.

First, I am enthusiastically appreciative to my project supervisor, Prof Maylor Leung Kar Hang, whose guidance and supervision motivate me from the beginning of the project to the conclusion of the project. His supervision and experiences enables me to have a better idea and understanding on computer vision field throughout the whole process of developing this system. Prof Leung is very kind in supervising student that he still allocates time for all his students when he is very busy. Thank you Prof Leung for all the efforts and time you have been spending with me throughout the whole project.

Furthermore, I would like to take an opportunity to thanks my project Moderator, Mr Leong Chun Farn who also assisted me in developing the projects by offering me an opportunity to attend his lecture and practical session on computer vision. I sincerely appreciated the offer Mr Leong although I could not make it. I am eternally grateful on all the help offered by Mr Leong.

Besides that, I would like to thank my parents for their dedication and the many years of supports during my studies.

Last, but not least, I would like to thanks Utar and all my friends for supporting me during the FYP projects development.

## **Abstract**

In the advancement of technology nowadays, there is a small group of people who are not able to make fully utilization of the technology – the disabled. Thus, there is an interest to invent assistive technology to help this group of people.

In this paper, the author proposes an application to assist the visually impaired to find an object by exploring the computer vision technique. In order to achieve the goal of the application, **context based template matching** technique will be investigated and implemented to detect, track and guide a user's finger. Furthermore, due to the limited time frame on this project the targeted object is assumed to have a fixed colour tone to simplify the problem.

As the first step, the system will require user to manually crop one of his/her fingers from an image. Then the system will extract a contextual template from the finger image. After that, the system will track the user's finger by matching the template to the input image. Finally, the system gives instruction to guide the finger to the targeted object in the form of voice commands.

The final deliverable will enable a machine and its operator (in this case the visually impaired) to successfully guide and assist user's finger to reach targeted object by simple voice instructions.

## TABLE OF CONTENTS

<b>TITLE</b>	<b>i</b>
<b>DECLARATION OF ORIGINALITY</b>	<b>ii</b>
<b>ACKNOWLEDGEMENTS</b>	<b>iii</b>
<b>ABSTRACT</b>	<b>iv</b>
<b>TABLE OF CONTENTS</b>	<b>v</b>
<b>LIST OF FIGURES</b>	<b>viii</b>
<b>CHAPTER 1 : Introduction</b>	<b>1</b>
1.1 Introduction	1
1.2 Motivation and Problem Statement	3
1.3 Project Scope	4
1.4 Objectives	5
1.5 Impact, Significance and Contribution	6
<b>CHAPTER 2 : Literature Review</b>	<b>7</b>
2.1 Mobile System to locate lost item for the Visually Impaired	7
2.2 CrossGuard	9
2.3 Smart Indoor Navigation	11
2.4 The GuideCane	13
2.5 Dishthi ( Integrated Navigation for Visually Impaired )	15

<b>CHAPTER 3 : Methodology &amp; Technology</b>	<b>17</b>
3.1 Methodology	17
3.1.1 Initialization	18
3.1.2 Finger Detection	22
3.1.2.1 Creating Skin Filter	24
3.1.2.2 Detecting Parallel Lines (with skin colour)	25
3.1.2.3 Scaling and Rotating Template Image	28
3.1.2.4 Template Matching	30
3.1.2.5 Compute Matching Percentage	31
3.1.2.6 Obtaining Fingertip Location	32
3.1.3 Object Detection	33
3.1.4 Compute Direction	35
<b>CHAPTER 4 : Performance Overview</b>	<b>36</b>
4.1 Overview	36
4.2 Performance Analysis	37
4.2.1 Different Lighting Condition	37
4.2.2 Type of Background	42
4.2.3 Numbers of fingers	44
4.3 Limitations of the system	45
4.4 Performance Results	46
<b>CHAPTER 5 : Project Review</b>	<b>60</b>
5.1 Conclusion	60
5.2 Improvements and Recommendations	61
5.3 Future Work	62
5.3.1 Object Detection	62
5.3.2 Further improvement on Finger Detection	62

<b>REFERENCE</b>	<b>63</b>
<b>APPENDIX A</b>	<b>A-1</b>
A-1 Project Poster	



## LIST OF FIGURES

<b>Figure</b>	<b>Title</b>	<b>Page</b>
Figure 2.1a	Overview of interaction flow of the system	8
Figure 2.3a	Horizontal Ceiling Structure and Passing Direction Edges	12
Figure 2.4a	How GuideCane works	14
Figure 3.1a	General Block Diagram	18
Figure 3.1.1a	Block Diagram – Initialization Module	20
Figure 3.1.1b	Manual initialization by using mouse	21
Figure 3.1.1c	Showing the pair of longest straight line being captured in the ROI	21
Figure 3.1.1d	Template image saved for matching purpose	22
Figure 3.1.2a	Block Diagram – Finger Detection	24
Figure 3.1.2.2a	HoughLine Standard vs HoughLine Probabilistic	27
Figure 3.1.2.2b	Pixel that is selected for skin colour test	28
Figure 3.1.2.6a	Obtaining fingertip location	33
Figure 3.1.3a	Block Diagram – Object Detection	35
Figure 3.1.4a	An example of navigating finger to a red colour object	56
Figure 1a	Finger tracking – Best Light	39
Figure 2a	Finger tracking – Medium Light	40
Figure 3a	Finger tracking – Low Light (fail)	41
Figure 3b	Finger tracking – Low Light (success)	41
Figure 4.2.2a	Finger detection on uniform background	43
Figure 4.2.2b	Finger detection on clustered background	44

<b>Figure</b>	<b>Title</b>	<b>Page</b>
Figure 4.2.2c	Finger Detection on skin colour background	44
Figure 4.2.3a	Finger Detection ( 5 fingers )	45

## **Chapter 1 Introduction**

### **1.1 Introduction**

Technology is a very important part of life. As technology advances either in memory or processing power with ever reducing price, it accelerates the growth of computer vision technology. In the history of computer vision development, people in the past faced plenty of problems, and now part of them can be solved. This technology advancement makes real time and realistic application a possibility.

Today, there is a small portion of people unable to fully utilize the technology for example people with visual impaired problem. There are many ways to fully utilized technology to benefit this group of people. In this project, the author intends to employ camera software to assist and guide the visually impaired to locate for specific object. A user friendly assistive human computer interface enhanced with computer vision technology will be the deliverable for this project and the software will be able to support the visually impaired to localize and to pick up object.

The recognition of finger is one of the main issues of some computer vision software to assist human. Unfortunately, different users have different fingers, for example finger size, length, skin colour and finger feature. Besides that, the background can be complicated by background with colour similar to that of human skin. Furthermore, there are circumstances that the user rotates his/her hand in depth to reach the object, and the software might lose track of the user's finger. Therefore, an appropriate approach will need to be chosen to eliminate these problems.

The project is mainly divided into two main parts – the object detection and finger tracking. The technique used to perform finger tracking is – context based template

## Chapter 1 : Introduction

matching. Context based template matching seems to perform better compared to other approach such as invariant features point detector approach in the case of accuracy and simplicity. Invariant features point detector is a good algorithm to track feature point in an input image provided the number features point must achieve certain level. Unfortunately, the features point extracted from the finger is way below the level with respect to scale invariant, thus invariant features point detection cannot solve the problem. On the other hand, context based template matching can outperform invariant feature points detection by only matching a binary edge template image of a finger to the edge input. In order to have an invariant scale and affine feature on template matching, the system dynamically scale and rotate the template image according to parallel straight line detected on the input. Thus, context based template matching method will have the ability to handle different scales and rotations.

The main objective of this project is to enable the visually impaired to pick up and locate an object. Due to the limited time of this project, the system will assumed that the object have a fixed colour tone and the object can be easily obtained by simple HSV thresholding method.

After the locations of the finger and object have been obtained, the system will issue simple voice commands to assist and guide the finger to reach and grab the object.

## **1.2 Motivation and Problem Statement**

Lots of computer vision related researches had been done for the visually impaired to solve their sight related problem. The motivation of this project is to develop an object finder for the visually impaired since they are having difficulties locating and finding an object without sight. There is a need and responsibility for the society and community to explore technologies to help the visually impaired.

According to (Robin & Arlene, April 2002), the population of visually impaired is growing at an alarming rate. These, coupled with the availability of inexpensive digital cameras and computers have motivated the author into developing Object Finder to assist visually impaired to locate for items.

### **1.3 Project Scope**

Object Finder for the visually impaired system is to be designed in a way to ease visually impaired to navigate his/her finger to reach and grab an object. Thus, the system will be implemented by utilizing computer vision technology – OpenCV library.

In order to achieve the goals of guiding finger of the visually impaired to the targeted object, the system will need to locate the finger position and locate the object position. By identifying the two locations, a direction can be easily computed and navigation process can be easily carried out as well.

This project has a larger weight allocated on the finger detection module compared to object detection module. Different computer vision approaches will be tested on the finger detection and the best solution will be identified.

On the other hand, the object detection module is simplified in the way that the system will assume the targeted object with fixed colour tone. The object detection part will be further enhanced in future due to time constraints.

Thus, the final deliverables of this project is a prototype to assist visually impaired to navigate his/her finger to reach and grab a predefined colour tone object ( For example, a red object ) with voice commands.

## **1.4 Objectives**

The system from this resulting project is to enable a visually impaired to find the object using image processing technique with the help of web camera and a computer. It must be able to perform actions like :

1. Able to get real time input from the camera.
2. Able to detect and track user's finger
3. Able to locate targeted object based on colour
4. Maximize the performance and accuracy of detecting user's finger
5. Perform simple guidance to assist the visually impaired to grab an object.

## **1.5 Impact, significance and contribution**

With the final deliverables of this project, the system will be able to assist the visually impaired to locate the item they need. This system helps this group of people to become more independent. They can locate object that they want rather than depending on other people to help them. This might indirectly help the family and the society to have cost saving spent on visually impaired.

In addition, the system also increases the confidence level and the happiness level of the visually impaired in the way that they became more independent.

Besides that, this system can also be used by human like robotics which can act and behave the same like human. The navigation part of the robotic hand will have the same problem as this project. Thus, the solution in this project can also be implemented in the robotics system.

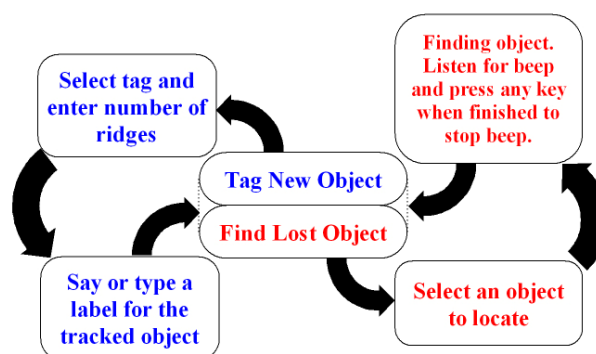


## **Chapter 2 : Literature Review**

### **2.1 - Mobile System to locate lost item for the Visually Impaired**

The visually impaired pedestrians trust on accessible infrastructure, technological aids, and specialized training. Technological aids can accompaniment both accessible infrastructure and techniques like the work by (Julie, A. Kientz; Shwetak, N. Patel; Arwa, Z.Tyebkh, 2006).

This system is an application running on mobile phones with build in Bluetooth technology to keep track of one object with Bluetooth tag attached. This approach is more simple and easy to implement compared to image processing for object detection which deals with difficulties such as background noise and others factor. The Bluetooth tag automatically emits waves from the tag so that the mobile phone can detect them. The disadvantage of this approach is that the Bluetooth tag that attached to the object is powered by battery. When the battery deflates, the detector system can no longer receive signal from the tag anymore, thus making the object finder failed to work.



**Fig 2.1a Overview of interaction flow of the locator system**

## Chapter 2 : Literature Review

The Bluetooth tag identifies itself using the MAC address on the Bluetooth chip. Thus, every tag has different identity. The visually impaired had to tag/attach the Bluetooth tag to the targeted object (in physical) and enter the tag's identity (in digital) into the system. User can also use voice to control the system to locate for the object. Whenever the user wants to find an item, the user can press the keypad on the phone or interact with the system through voice recognition (Fig 2.1a). Then the system will send signal to the Bluetooth tag so that they will "beep". The user will be able to follow the "beep" sound to locate the item. After the user have reach the item, he/she presses the keypad once again to stop the "beep" sound.

## **2.2 - CrossGuard**

The CrossGuard system (Truong, 2012) is a navigational aid for visually impaired resident. The authors realized that visually impaired pedestrians experienced problems and challenges when they are navigating outdoor. Visually impaired will need more information when they are navigating in area that they are not familiar with, especially on the road intersections.

The system gets the current coordinate of the user in real time with the help of GPS technology. The system is preinstalled with map data from Google Street View, OpenStreetMap (OSM). This will enabled the system to provide “sidewalk to sidewalk” direction for the visually impaired. Besides that, the system will be able to help the user to understand intersection geometry by including information about the size and shape of the streets that meets in the intersection as well as details about the traffic signaling.

First, the user must input their desired destination into the system through existing input technique such as keyboard or the build in voice recognizer. Then the system will generate a route based on the information from Google Street View and Open Street Map (OSM). The system put emphases on geometrical information of the sidewalk such as location of streets and the angle of which street intersect with which street will be stored in the system’s database throughout the navigation process. The system will feedbacks to the user via an audio system.

During the navigation process, user will be able to interact with the system through some simple gesture on a touch screen devices such as mobile phone. User will need to perform tapping gesture on a touch sensitive device which serves as an input to the system. User will be able to interact with the system with different type of hand gestures such as swiping left or right, double tapping on the touch sensitive device to ask

## Chapter 2 : Literature Review

predefined questions to the system. The limitation of this system is user will not be able to ask different question other than those which is already defined in the system.

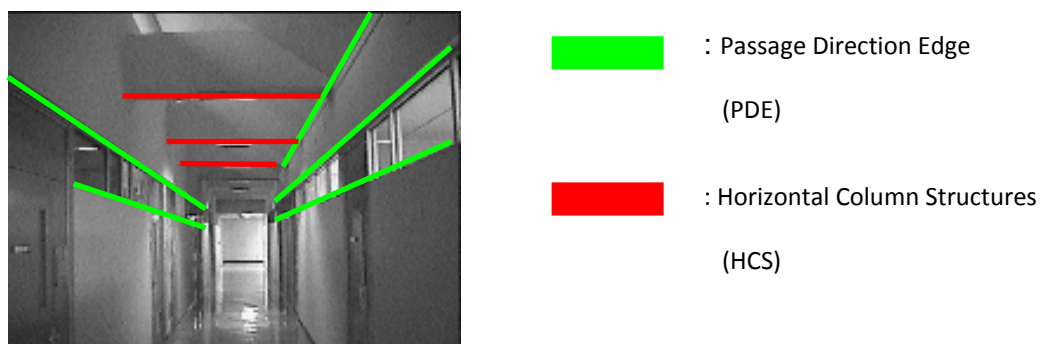
This system is highly dependent on GPS system which mostly works outdoor. In order to navigate indoor, the system will need to find alternative way to overcome the GPS signal issue. Then the system will be able to work perfectly.

## **2.3 Smart Indoor Navigation**

In the study of computer vision, Smart indoor navigation (Wee Ching & K. H, 2007) is a navigation tool designed for the visually impaired. This system utilized specific visual clues gathered from images captured by a light weight camera, mounted on a user. Each visual clue captured by the camera will be stored in a map database and they contained a unique identifying pattern template that will be used for discovery purposes. The system mainly collects its visual clue from the ceiling region for a successful navigation.

The system captures the positions of the edge by processing the image captured in real time from the light weighted camera through a series of image processing procedures. The positions enable the system to identify the basic visual clues in the image. For example, calculating the angle a line makes with the horizontal. The coordinates are then studied and analyzed by the system to detect common corridor structures. Then the information obtained from the analysis will be used to form decision to help in the navigation.

In the analysis stage, several clues must be detected first before the system can actually work correctly. They were the Horizontal Ceiling Structure(HCS) and Passage Direction Edges(PDE) (Fig 2.3a).



**Figure 2.3a – Horizontal Ceiling Structure and Passage Direction Edges**

## Chapter 2 : Literature Review

Template matching technique will be employed when the user comes to a turning points or a door entrance. An identifying pattern that uniquely identifies a door or turning point will be searched through a series of images. If the pattern is found, then the specified door or turning point is located.

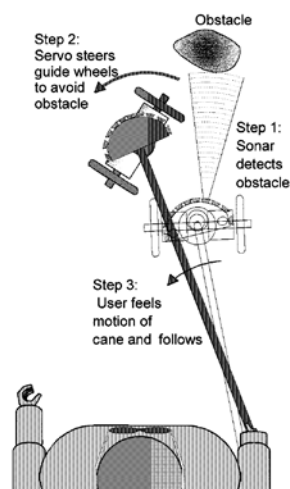
In order for Smart Indoor Navigation to be a complete system (not being implemented by the author), after the analysis process, maps were being generated and being stored in the database of a host computer through 802.11 wireless interface. Whenever a new visual clue is being detected, the system will intelligently transfer the information to the server via the wireless interface to the host computer and update the current position of the user in the map. The host computer will then prompt the device which visual clue to detect next.

The SIN system basically only focuses on road navigation for visually impaired but fail to address the issue that the visually impaired will need guidance when they are locating some object. SIN system does not provide a complete navigation that the visually impaired desire to.

## **2.4 The GuideCane**

The GuideCane (Iwan & Johann, 2001) is a novel device to help people with visual impaired problem to navigate in a fast and safe manner by avoiding obstacles and other hazards. The GuideCane is an enhanced version of the White Cane ( normal travel aid for the blind ) stick equipped with ultrasonic sensors to detect obstacles. GuideCane will provide the user feedback by steering actions to help visually impaired to avoid the obstacles.

GuideCane is made up of a normal White Cane plus an intelligent system integrated with the cane itself. With the aid of two rollers, GuideCane can provide the feedback to the user with the direction they should move when there is an obstacle in front. For example if there is an obstacles on the path where the visually impaired will pass, the GuideCane actually detect the obstacles first, then the two rollers will turn in a direction to avoid the obstacles (shown in Fig 2.4a). Thus, the visually impaired will immediately feel the change and avoid the obstacles by walking towards the direction the roller turns to.



**Fig 2.4a – How GuideCane works**

## Chapter 2 : Literature Review

GuideCane is a creative idea that to assist visually impaired when they are travelling around. GuideCane system which is integrated into the Process Control Board (PCB) of the GuideCane is an intelligent piece of software. The system actually decides what action and feedback to be performed in order to avoid the obstacles. Thus, GuideCane does not require any conscious effort on its user.



### **2.5 – Dishthi ( Integrated Navigation for Visually Impaired )**

Dishthi (Abdelsalam; Balaji; Steven, Edwin ,2001) – a navigation system developed for the visually impaired is a wireless navigational system. It is built up of different kinds of components including voice recognition and synthesis, Global positioning system, wearable computers, wireless networks and Geographic Information System. The environmental conditions will be queried from a database along the user's current route through the use of GPS and GIS, the details are then given to the user through voice cues. For example, the path information is delivered to the user in real time through the voice cues. Furthermore, the system will have the ability re-route if the user decides to change destination and is able to take note from the user about certain condition.

This system provides the user with augmented contextual information based on the user's preference, contextual constraints and obstacles that are dynamic. In other words, the system is able to navigate the visually impaired through static and dynamic paths. This system is created to supplement other navigational tools such as canes, wheel chairs and even blind guide dogs. Currently, this system is able to provide the user with a preferable route as the shortest route may not be the best route for a visually impaired person because it may not have the least hazards.

In addition, the system is developed with GPS, GIS and other integrations because a route is not always static. Therefore, a visually impaired person cannot rely only on their regular or repetitive routes. For example, a route may be obstructed by unexpected natural hazard or a wet puddle which appears after a big rain. To rely on traditional navigational aids such as canes, the user will not be able to detect unexpected obstacles such as a rock in front of them and this may lead to an accident. However, with Dishthi, the user will be able to avoid such unwanted scenarios because they will be alerted before they run into such obstacles.

## Chapter 2 : Literature Review

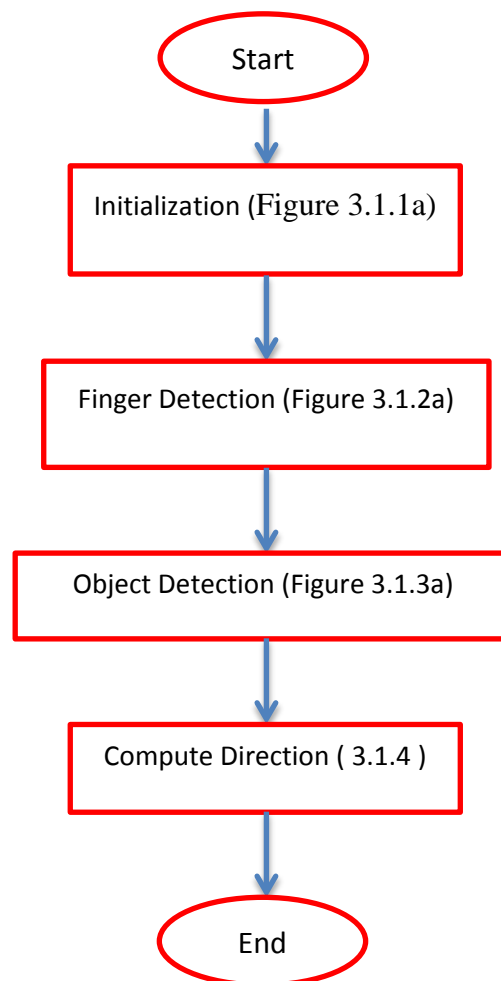
This system is closely related to augmented reality as the virtual environment is modeled using a GIS database while the user's location is obtained through the GPS system. The main goal of the developed system is to provide the visually impaired with enough information so that they can walk comfortably from one location to another by taking in contextual factors and unexpected factors such as road blocks.

Furthermore, this system can be further extended to support other applications such as routine building maintenance by physical plant crew and even for emergency response system which is suggested by the authors. Moreover, this system can also be used by a normal person when they are navigating through an unknown environment or a dark contextual environment.

## **Chapter 3 Methodology & Technology**

### **3.1 Methodologies**

In this Object Finder system, image frames are captures through a web camera. Then, the image will be passes through three main image processing module – **initialization, finger detection, and object detection**. The general flow chart of the entire system is shown in Fig 3.1a.



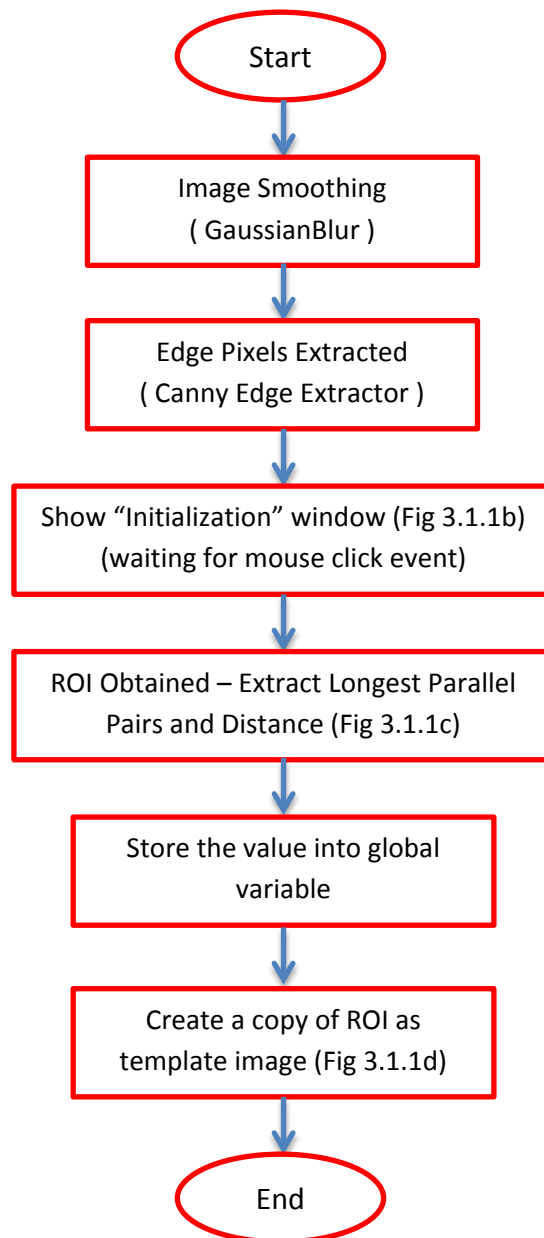
**Fig 3.1a – General Block Diagram**

### 3.1.1 Initialization

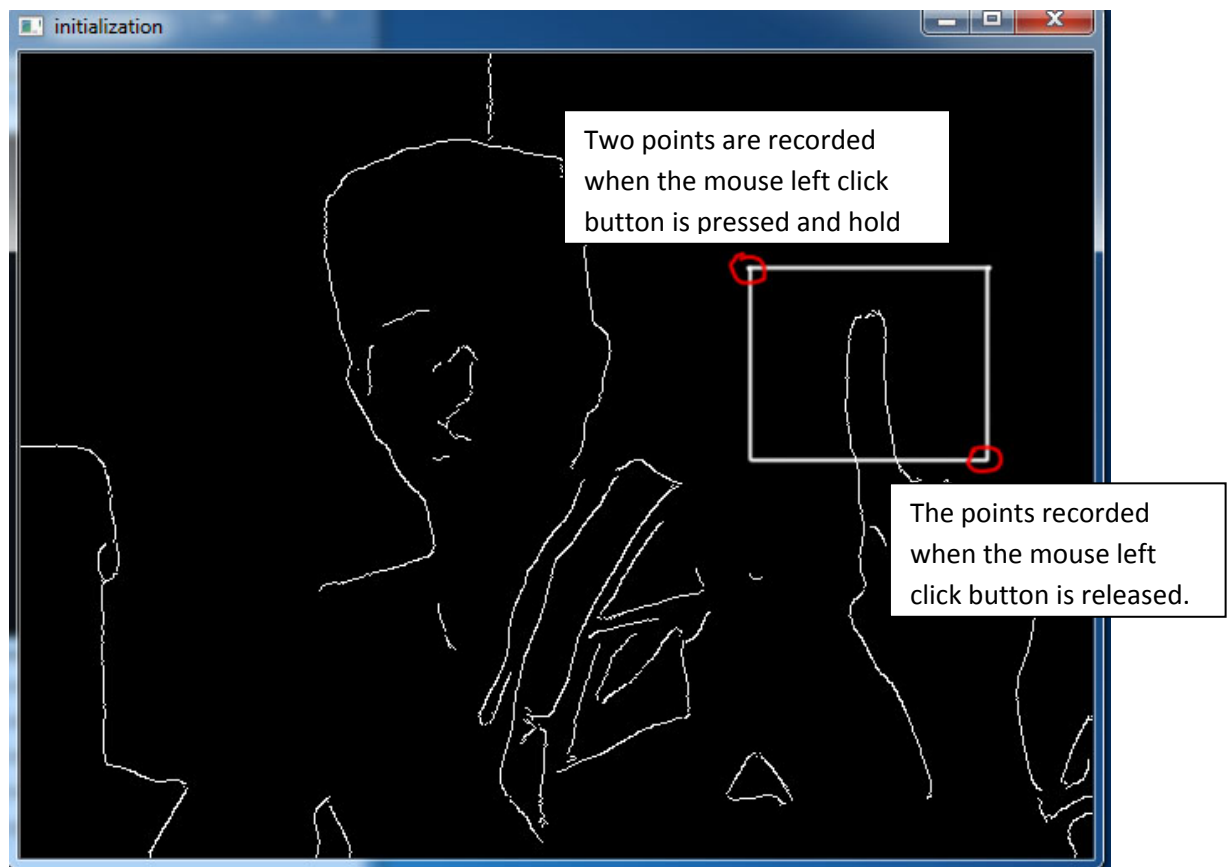
In the initialization module, user will need to manually capture user's finger with the input from a mouse (shown in Fig 3.1.1b). The input from the webcam will undergo image smoothing function provided by OpenCV library which is GaussianBlur with **sigmaX value of 3.0, sigmaY value of 0.0**. The choice of sigmaX 3.0 and sigmaY 0.0 values is chosen because it yields the best result in reducing noise from the image. The chosen control parameter is used to generate all results.

Next, edge pixels are being extracted from the processed input image using Canny Edge Detector. **The low threshold value is 25 and the high threshold value is 75**. With this two threshold value, the edge pixels extracted are satisfactory for most of the conditions. Thus, the values for Canny Edge Detector will be also populated across the system. Then the edge matrix will be showed to the user as an "initialization" window.

User will have to draw a rectangle on the "initialization" window to select the region of the finger part. Then the system will automatically obtain the longest parallel straight lines, the distance between the parallel straight lines (to calculate scale) and the finger orientation ( $\theta$ ) (shown in Figure 3.1.1c). The image in the rectangle drawn will be store as a template image (shown in Figure 3.1.1d) for further processing in finger detection module. The simple flow chart of the initialization module is shown in Fig 3.1.1a.



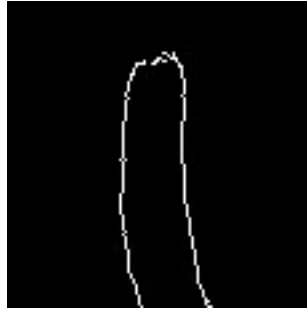
**Figure 3.1.1a – Block Diagram of Initialization Module**



**Fig 3.1.1b – Manual initialization by using mouse**



**Fig 3.1.1c – Showing the pair of longest straight line being captured in the ROI**

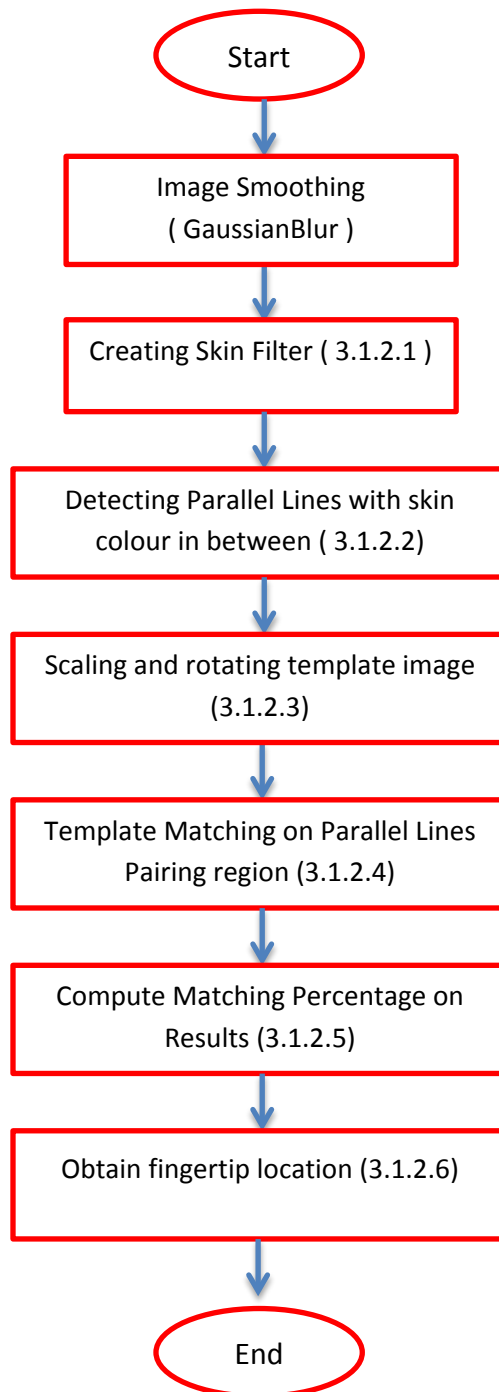


**Fig 3.1.1d – Template image saved for matching purpose later**

### 3.1.2 Finger Detection

The general flow of finger detection module can be further breakdown into creating a skin filter, detecting parallel line in the input image (with skin colour between the parallel lines), scaling and rotating template image, template matching of all pair of parallel lines, computing matching percentage, and selecting the best matched as result. Figure 3.1.2a shows the general block diagram of the finger detection modules.





**Fig 3.1.2a General Block Diagram of Finger Detection**

### **3.1.2.1 Creating Skin Filter**

Although there are a lot of methods to create a skin filter out there, the most commonly used technique is through HSV thresholding method. Thus, the author selected the HSV thresholding technique to create the skin filter. According to (Garcia, 1999), a normal human skin colour has the range of HSV values as follows.

$$0 \text{ degree} \leq H \leq 50 \text{ degree},$$

$$20\% \leq S \leq 68\%,$$

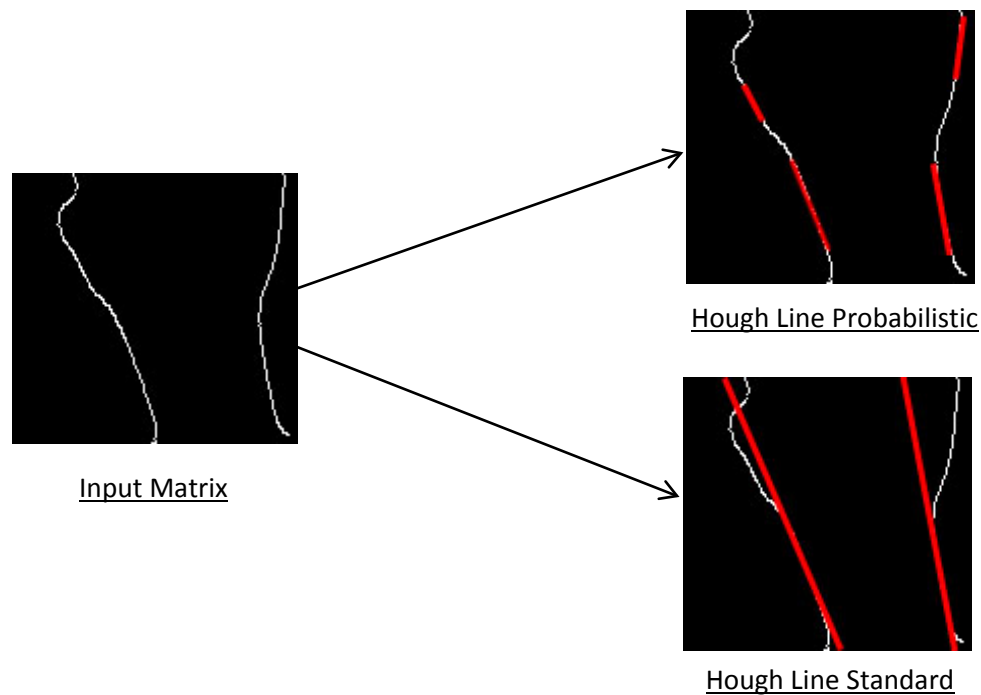
$$V \geq 35\%,$$

Thus, the skin filter of the Object Finder system will be created using these values. The result obtained from this skin filter is quite accurate and satisfactory. The matrix will be stored as skin mask after thresholding the input with these values.

### **3.1.2.2 Detecting Parallel Lines with Skin Colour Between The Lines**

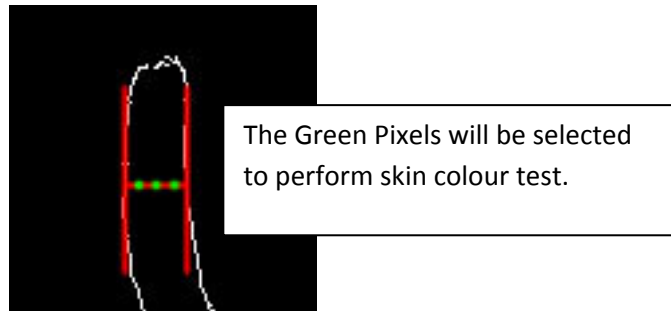
Edge pixels are being extracted from the web camera and the threshold values will be the same as the initialization stage. After all edge pixels had been collected, Hough Line Transformation (Probabilistic) function will be used to extract all the possible straight line in the matrix. **Hough Line Probabilistic** method is chosen over the Standard version because we need to extract lines in segments. The difference between Hough Line Standard and Hough Line Probabilistic will be shown in Fig 3.1.2.2a.

After all the straight lines is being collected in an vector, the straight line vector address is then passed into a function ( `locateParallelLines()` ) to extract all the parallel lines in all the straight line extracted previously. In order to define two lines are parallel, the difference between theta  $\theta$  values of the two lines must be equal to 0. But in this case, the system considered two lines as parallel when the **difference between theta  $\theta$  values is below 20** due to the structure of the finger. Different people have different type of fingers; For example, some people might have a sharp and thin finger. If we restrict the difference of theta to 0, then people with sharp and thin finger will not be able to utilize the system.



**Fig 3.1.2.2a Differences between HoughLine Standard and HoughLine Probabilistic**

After all the parallel pairs is computed, the system will iterate all the pair to locate for skin pixels between the midpoint of parallel line's midpoint. 3 points will be selected for the skin colour test to filter out candidates (shown in Fig 3.1.2.2b). The 3 points will be tested using the skin mask that we created previously. All the parallel pairs that have the 3 points with skin colour positive (e.g Fig 3.1.2.2b) will be selected to perform template matching in the next process.



**Figure 3.1.2.2b – Pixels that is being selected for skin colour test**

### **3.1.2.3 Scaling and Rotating Template**

As we all know that template matching is not invariant to scale and affinity, which is the reason that we have to scale and rotate the template explicitly according to each pair of parallel lines.

#### **Scaling**

For the remaining parallel line pairings, the distance of the two parallel lines is calculated as ( $D_{input}$ ). Let the width of the template be ( $D_{template}$ ).

Thus the scale factor will be :

$$\text{Scale Factor} = \frac{D_{input}}{D_{template}}$$

After scale factor is being computed, the template image will be scale using the `resize()` function provided by the OpenCV library with the values correspondingly.

**Rotating**

Besides scaling the template, the rotation of the template will also be an important step for template matching to match perfectly. Let the orientation of the template be  $(\theta_{\text{template}})$ . Thus, the template image will need to be rotated according to the orientation  $(\theta_{\text{input}})$  of a pair of input parallel line.

The rotation angle can be computed as :

$$\text{Rotation (in degree)} = \left\{ \begin{array}{ll} 180 - ((\theta_{\text{template}}) - (\theta_{\text{input}})) & \text{where } \theta_{\text{template}} = -ve \text{ and } (\theta_{\text{input}}) = +ve \\ - (180 - ((\theta_{\text{template}}) - (\theta_{\text{input}}))) & \text{where } \theta_{\text{template}} = +ve \text{ and } (\theta_{\text{input}}) = -ve \\ (\theta_{\text{template}}) - (\theta_{\text{input}}) & \text{otherwise} \end{array} \right.$$

### **3.1.2.4 Template Matching**

First, the system will allocate a matrix reference to have the area of the parallel lines pairing, and then template matching technique (OpenCV Community, 2013) will be performed on the ROI. The template image (T) will be slid through the source image (I). By sliding, it means moving the patch one pixel at a time (left to right, up to down). At each location, a metric is calculated so it represents how well the match at that location. The highest matching location can be calculated by using the MinMaxLoc function on the metric. The template image (T) will be obtained from the previous module (scaling and rotating). The matching function will return the position where the template is best matched to the ROI.

The system does not use the classical template matching method to do the hand tracking, but a better version of it. For each parallel line pairs which remain after all the filtering, template matching will be performed on the specific region around the parallel line pairings to get the best match. For each parallel line pairing =  $\{P_1, P_2, P_3, \dots, P_n\}$  , the template result generated will be  $\{R_1, R_2, R_3, \dots, R_n\}$ .

In order to select the best matching among all the  $R_n$  , there will be a demand for the system to calculate the matching percentage between the result and the template (covered in 3.1.2.5).



### **3.1.2.5 Compute Matching Percentage**

In reference to 3.1.2.4, the system will generate a result set of matching  $\{R_1, R_2, R_3, \dots, R_n\}$ . In order to select the best among  $R_n$ , we have to compute the matching percentage to determine how well the template had been matched. The result that has the highest percentage of matching will eventually have the highest possibility to be the user's finger.

The matching algorithm:

T – Scaled and Rotated Template (from 3.1.2.3)

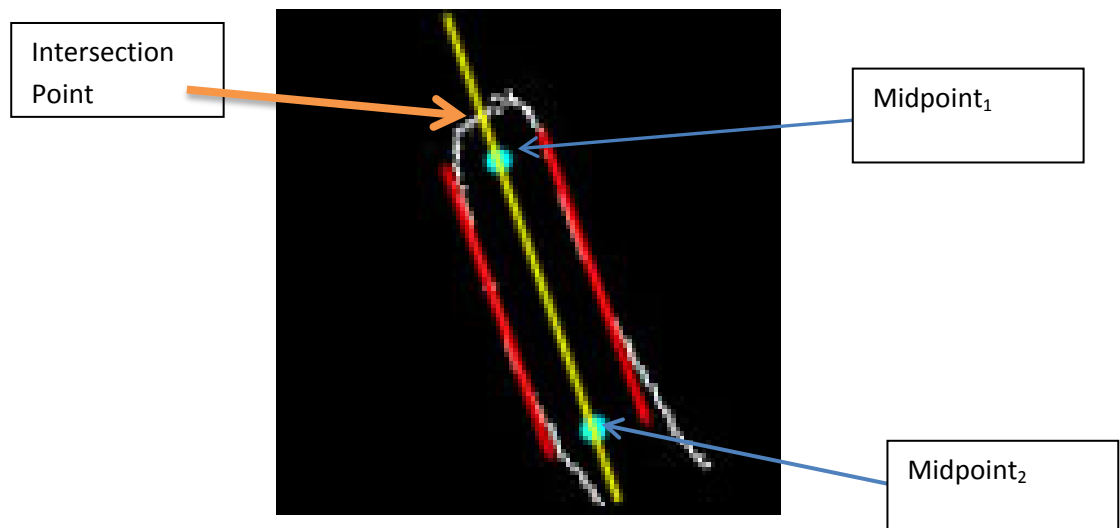
$R_n$  – Result Matrix generated from 3.1.2.4

$$\text{Matching rate} = \frac{| \text{T and } R_n |}{| \text{T} |}$$

The maximum matching rate will be 1 or 100%, which means that the result is at the perfect location with perfect orientation and scale. This on the other hand means the higher the matching rate, the higher the probability that the region is the user's finger. Thus, the aim of this function is to calculate all the matching rate for the result  $\{R_1, R_2, R_3, \dots, R_n\}$  and locate the maximum one.

### 3.1.2.6 Obtaining Fingertip Location

The result that has the maximum value of matching will be process in this function. The basic idea of this function is to get the location of the fingertip. In order to achieve that, a straight line is needed. Two midpoint set  $\{Mid_{x1} , Mid_{y1}\}$  and  $\{Mid_{x2} , Mid_{y2}\}$  will needed to be computed. By generating a straight using the two midpoint set, any point that has an intersection with any pixel of the template will be recorded as the fingertip point (Fig 3.1.2.6a – Example of obtaining fingertip location ).



**Fig 3.1.2.6a - Example of obtaining fingertip location**

### 3.1.3 Object Detection

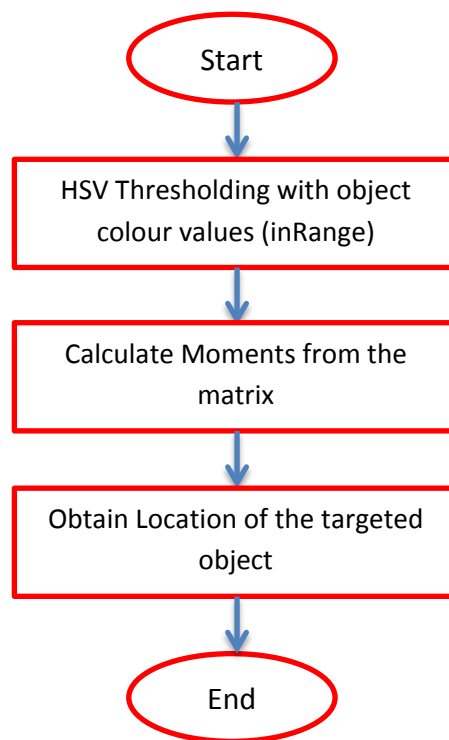
After the location of the fingertip had been determined, the system will also need to compute the location of the targeted object. Due to time constraint in the project, the author will assume that the targeted object is in pre-determined colour and it will be the only pre-determined colour object on the screen. The input image will undergo smoothing function and then converted to HSV colour space. HSV colour space is chosen because it actually separate colour components from intensity, thus it is more accurate when the lighting is different compared to RGB – the usual colour space. The block diagram of this module will be shown in Fig 3.1.3a.

Moments will be calculated on the threshold image to obtain parameters such as area, moment01, moment10, etc. From the moment01 and moment10, we can actually obtain the location of the object :

$$\mathbf{Pos}_x = \frac{moment10}{area}$$

$$\mathbf{Pos}_y = \frac{moment01}{area}$$

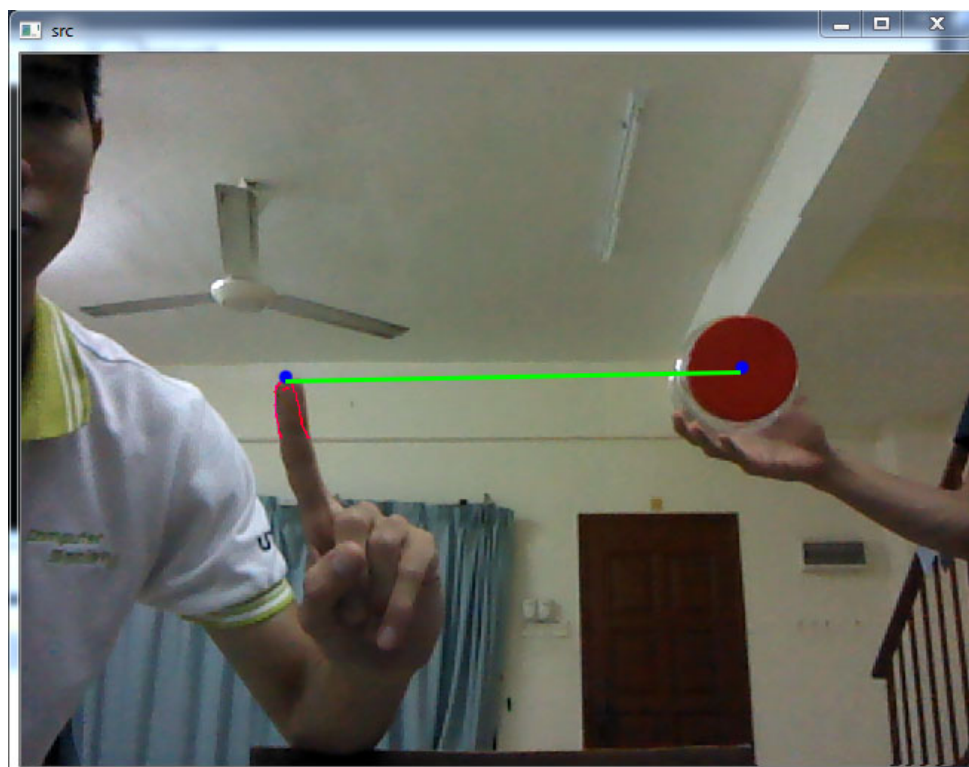
By having the location of the object, the system will be able to guide the user's finger to the object location.



**Fig 3.1.3a – Block Diagram of Object Detection**

### 3.1.4 Compute Direction

Once the two locations of the fingertip and the object have been obtained, then the system will draw a line connecting the two locations and compute direction based on the deviated angle. Assistive feedback such as voice command will be used to assist user on the direction the finger should move in order to reach the targeted object. Preconfigured wave sound will be played upon this stage as the voice command. Voice command module is running on top of another thread to prevent the main thread from interrupted. Figure 3.1.4a shows an example of drawing a line connecting the two locations.



**Fig 3.1.4a – An example of navigating finger to a red colour object**

## **Chapter 4 : Performance Overview**

### **4.1 Overview**

In order to test the accuracy of the finger detection module, the author has selected a few scenarios to test how well the finger detection can handle different conditions. They are categorized as 1) lighting conditions ( different level of brightness ), 2) background ( plain or clustered ), 3) The number of fingers ( 5 fingers ).

The objective of the performance test is to evaluate the accuracy of the finger detection module based on different conditions. To achieve a higher accuracy of finger tracking, edge pixel extracted must be very clear and obvious, providing that the lighting conditions must be sufficient for the skin colour to be distinguished from the background.

By using all the methods described in chapter 3, it has been found out that the system worked satisfactory in detecting and tracking finger, however, with some limitations which will be discussed later.

## **4.2 Performance Analysis**

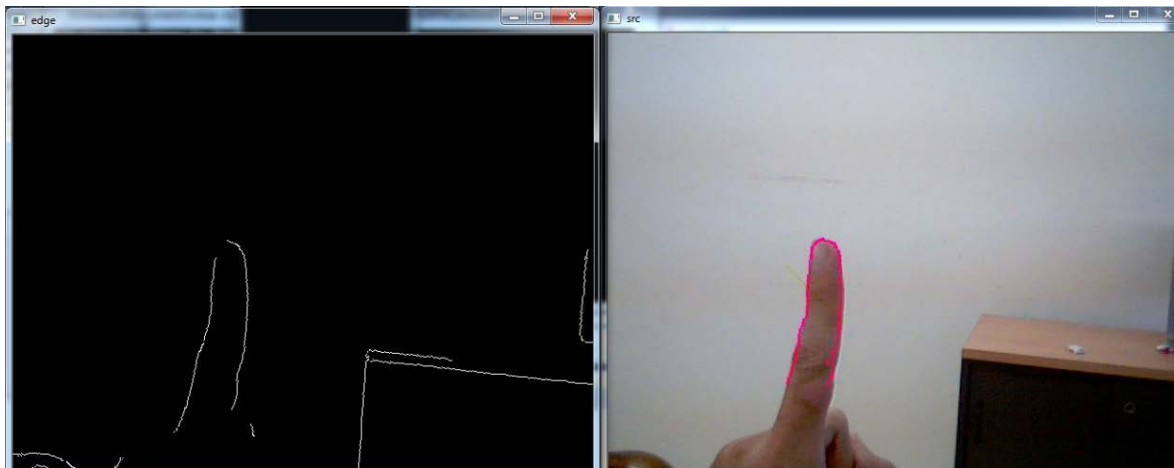
The system has achieved a quite high accuracy and consistency to detect and track user's finger. However, false detection and tracking may occur too. The factors that will affect the performance will be explained as follow.

### **4.2.1 Different Lighting Condition**

Three scenarios have been selected to test on the accuracy of the finger detection module. They are 1) Best Lighting, 2) Medium Lighting, and 3) Low Lighting. The finger detection module will be able to handle scenario 1 and 2 very well. The failure of the system depends on how low the light intensity and the background when the system is tested in scenario 3.

The hand detection module will function as long as the Edge pixel of the finger can be extracted without distortion. If edge pixels are not extracted in a clear and obvious manner, line detection will fail. Thus the system will ignore the region and the finger detection will fail as well.

**Scenario 1 – Best Lighting ( 2 Fluorescent bulb is switched on )**



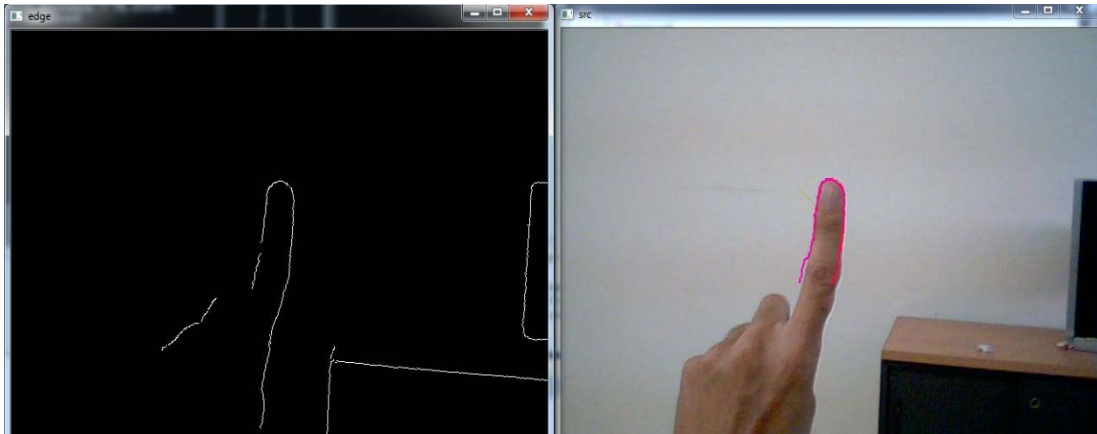
**Fig 1a – Finger tracking module under best light condition**

**Explanation :**

The edge can be clearly extracted from the input image thus, the finger can be successfully detected and tracked (shown in Fig 1a).



**Scenario 2 – Medium Lighting ( Only 1 Fluorescent bulb is switched on )**

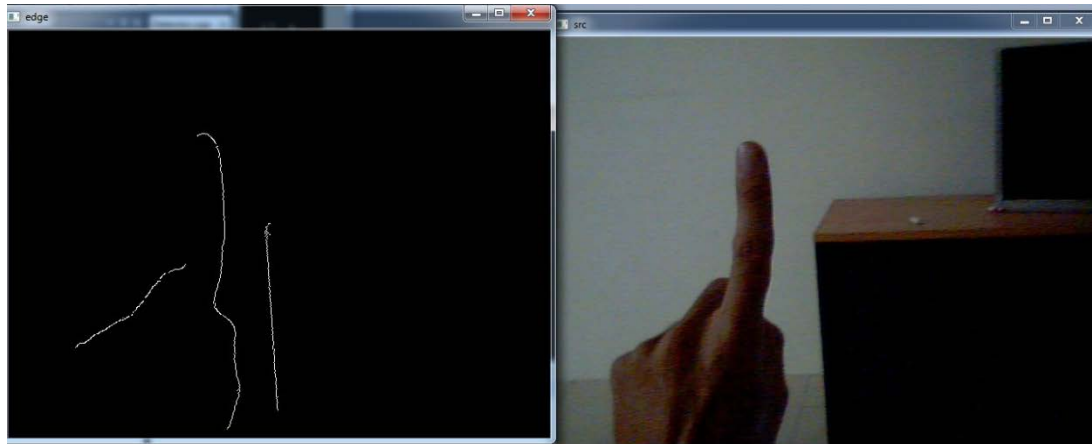


**Fig 2a - Finger tracking module under medium light condition**

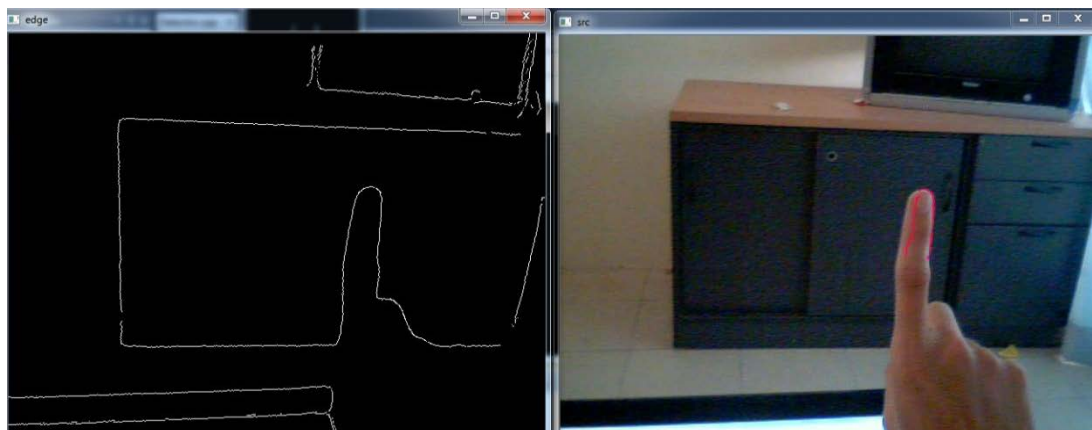
**Explanation :**

The finger can still be tracked successfully because the system are able to extract clear edges from the input image (shown in Fig 2a).

**Scenario 3 – Low Lighting ( Without any Fluorescent bulb )**



**Fig 3a – Finger Tracking Module on Low Light Condition ( failed )**



**Fig 3b – Finger Tracking Module on Low Light Condition ( successful )**

**Explanation:**

In Fig 3a, the system was unable to detect the finger because distorted edges were extracted from the input image due to low light intensity (shown in the left “Edge” window). But when the finger is moved to a different background (black uniform background) where the colour difference is quite high, then it will be detected successfully again ( Fig 3b ).

### **4.2.2 Type of background**

The system will be able to handle different type of background either uniform or clustered except when the background colour is similar to the skin colour. When the finger location is located on top of a uniform background (not skin colour), then the edge can be easily extracted provided the lighting condition is ideal ( shown in Fig 4.2.2a ).

On the other hand, the accuracy of finger detection module is also acceptable on clustered background. With this technique, the edge can still be detected easily due to the difference of the background colour and the finger colour (shown in Fig 4.2.2b).

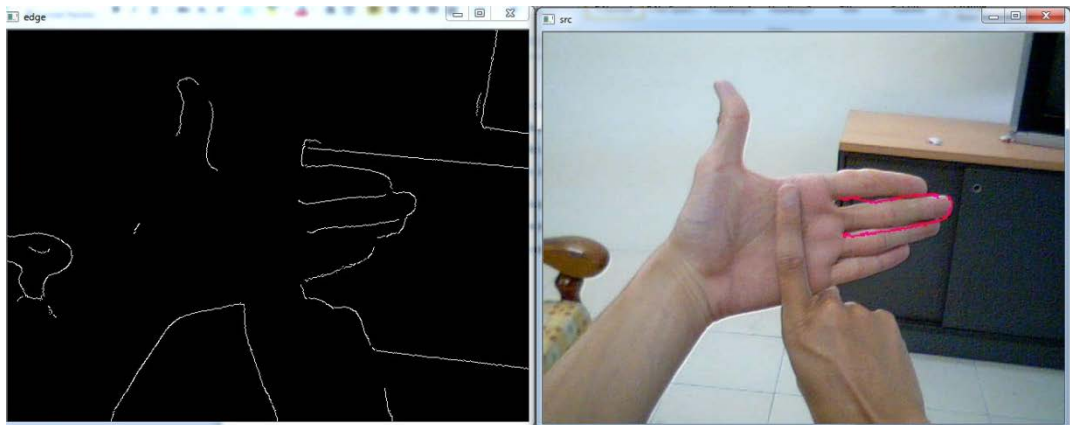
Unfortunately, when the background of the finger is made up of skin colour, the finger detection module will fail because the colour of the finger and the background is the same, thus no straight line or edges will be extracted (an example of this scenario is shown in Fig 4.2.2c).



**Fig 4.2.2a – Finger Detection on an uniform background**



**Fig 4.2.2b – Finger Detection on a clustered background**

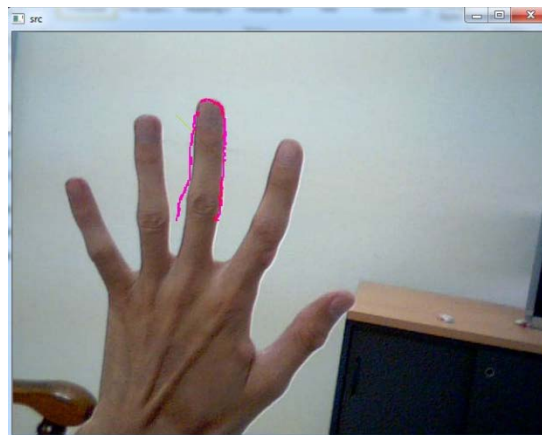


**Fig 4.2.2c – Finger Detection on a skin colour background (Failed)**

### **4.2.3 Number of fingers**

The accuracy of a finger got detected increases when the number of finger increases. This is because the probability of a parallel line pair to be detected as a finger increases. Thus by using more fingers in the system, the higher the chance of a finger finger to be detected (shown in Fig 4.2.3a).

For example, if the user has 5 fingers on the screen, while 3 of the finger edges are unable to extracted, the remaining 2 finger can still be detected as a finger and the navigation process can be executed.



**Fig 4.2.3a – Finger Detection module is still accurate on image with 5 fingers**

### **4.3 Limitations of the system**

This project has performed well and satisfied the requirement to detect and track user's finger. However, there are few drawbacks in the system like lighting conditions and different type of backgrounds.

Besides that, the current system is not fully automated yet – the initialization module. Visually impaired will need another people to actually initialize his/her finger into the system before he can use the system. Thus, in order for a fully automated system, the initialization will need to be redesigned in the future.

In order for the visually impaired to enjoy the full system of the Object Finder, the object detection part must also be further improved ( From simple colour detection to intelligent object detection module ). Unfortunately, limited time is allocated to this project, thus the object detection module will be focused in the future.

## **4.4 Performance Results**

A total of 50 images have been captured to test the performance of the system.

**They are :**

**Uniform background** – 20 images

**Clustered background** – 30 images

### **Result Summary :**





<b>Type of background</b>	<b>No of Image Finger Detected</b>	<b>Total Test Image</b>	<b>Success Percentage</b>
Uniform background	18	20	80%
Clustered background	23	30	76.666%



## Chapter 4 : Performance Overview

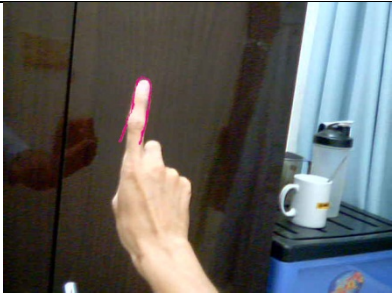
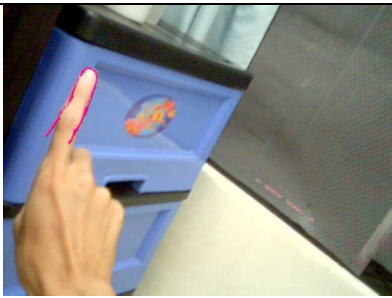
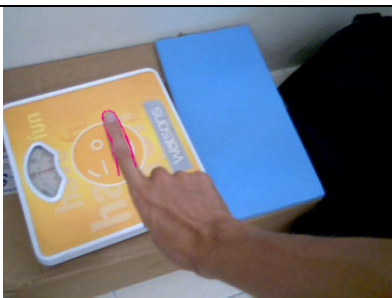
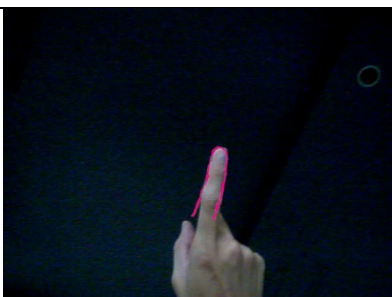
The results of the test are as follows:

### Uniform background :

No.	Sample Image.	Able to detect finger
1		✓
2		✓
3		✓
4		✓

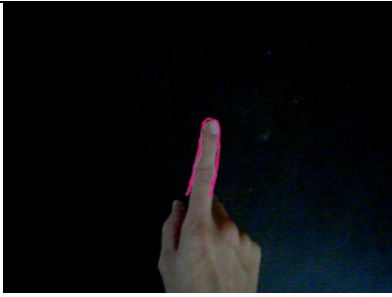


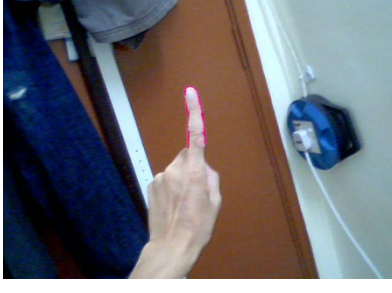
**Note :** The pink region shows the matching of the template to the finger.

## Chapter 4 : Performance Overview


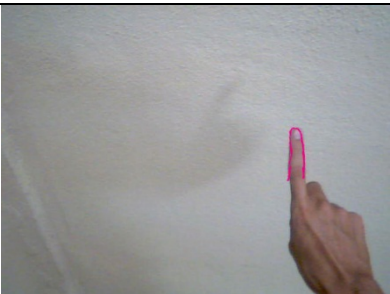
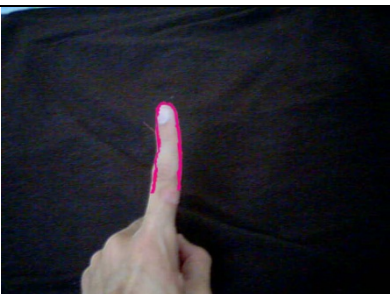

5		✓
6		✓
7		✓
8		✓

**Note :** The pink region shows the matching of the template to the finger.

## Chapter 4 : Performance Overview

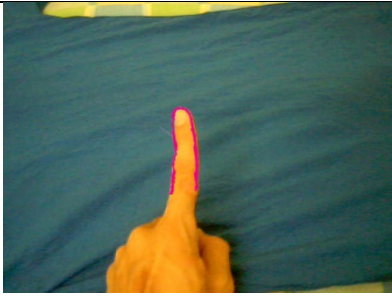



9		✓
10		✓
11		✓
12		✓

**Note : The pink region shows the matching of the template to the finger.**

13		✓
14		✓
15		✓
16		✓

**Note : The pink region shows the matching of the template to the finger.**

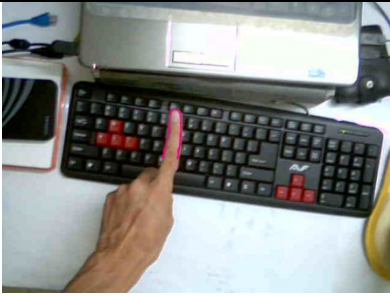
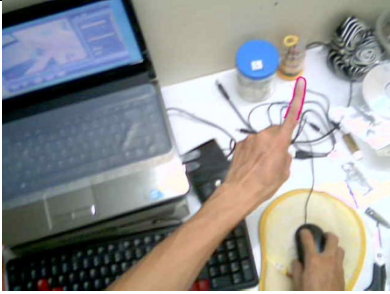

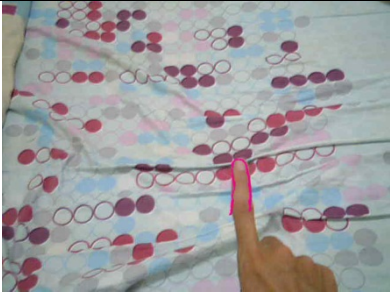
## Chapter 4 : Performance Overview

17		✓
18		✓
19		X
20		X

**Note : The pink region shows the matching of the template to the finger.**

## Chapter 4 : Performance Overview

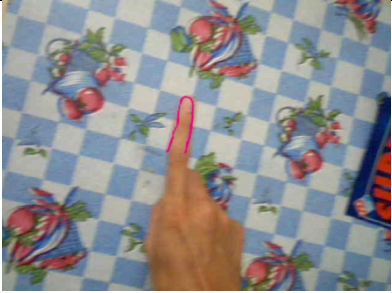


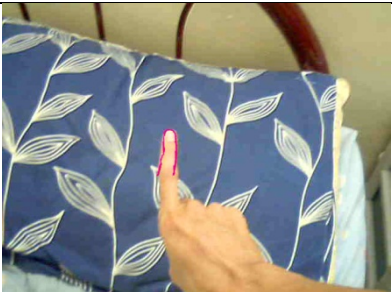
### Clustered background :

No.	Sample image.	Able to detect finger
1		✓
2		✓
3		✓
4		✓

**Note :** The pink region shows the matching of the template to the finger.



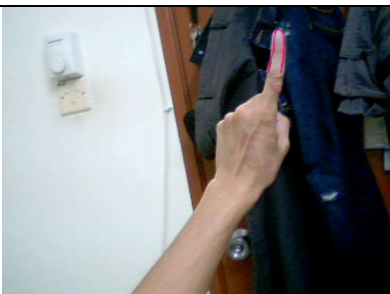



## Chapter 4 : Performance Overview

5		✓
6		✓
7		✓
8		✓

**Note : The pink region shows the matching of the template to the finger.**

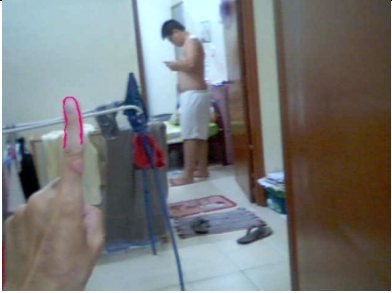
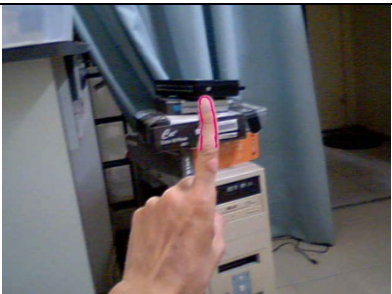

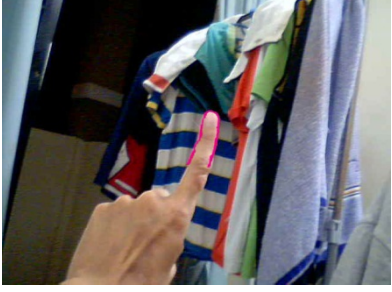
## Chapter 4 : Performance Overview

9		✓
10		✓
11		✓
12		✓

**Note : The pink region shows the matching of the template to the finger.**

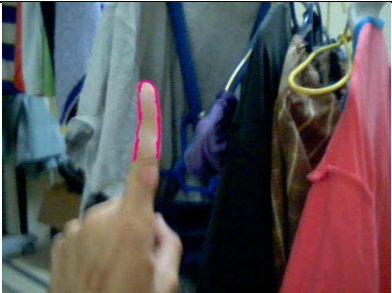
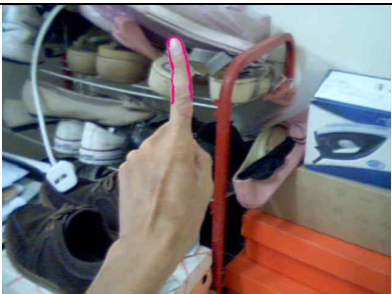




## Chapter 4 : Performance Overview

13		✓
14		✓
15		✓
16		✓

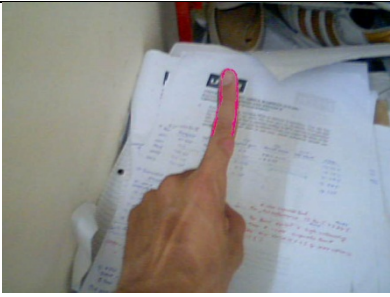

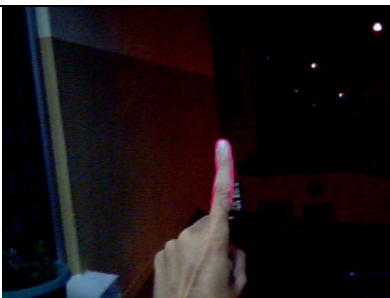
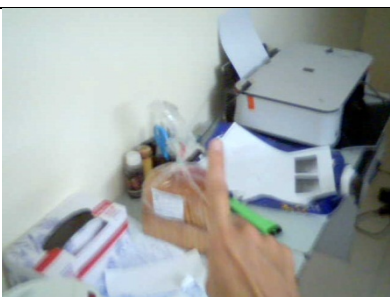
**Note : The pink region shows the matching of the template to the finger.**

## Chapter 4 : Performance Overview



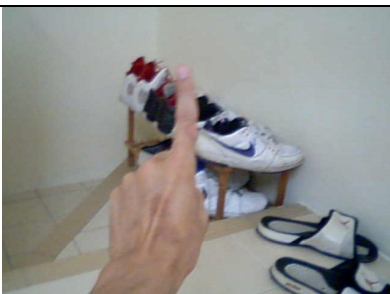
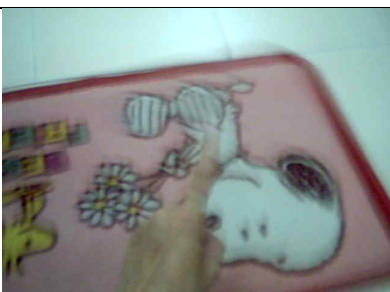
17		✓
18		✓
19		✓
20		✓

**Note :** The pink region shows the matching of the template to the finger.


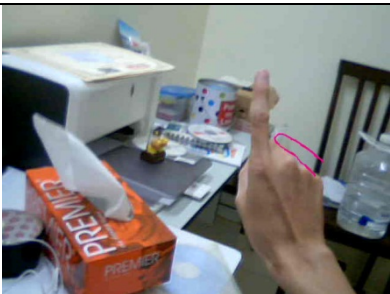
## Chapter 4 : Performance Overview

21		✓
22		✓
23		✓
24		X <b>Reason :</b> Finger's edge is not clear.

**Note :** The pink region shows the matching of the template to the finger.

25		<p><i>X</i> (not accurate)</p> <p><b>Reason :</b> Background is skin colour. Finger's edge is not clear.</p>
26		<p><i>X</i></p> <p><b>Reason :</b> Finger's edge is not clear.</p>
27		<p><i>X</i></p> <p><b>Reason :</b> Finger's edge is not clear.</p>
28		<p><i>X</i></p> <p><b>Reason :</b> Input image too blur.</p>

**Note :** The pink region shows the matching of the template to the finger.

29		<p><math>X</math> (not accurate)</p> <p><b>Reason :</b> Finger's edge is not clear.</p>
30		<p><math>X</math> (false positive)</p> <p><b>Reason :</b> Finger's edge is not clear.</p>

**Note :** The pink region shows the matching of the template to the finger.

## **Chapter 5 : Project Review**

### **5.1 Conclusion**

This paper has given an account of and reasons on the importance of delivering object finder software for the visually impaired using computer vision technique. The project was undertaken to design a user friendly object finder for the visually impaired to assist them in locating specific items employing voice commands.

In order to achieve the ultimate goals, different computer vision techniques - feature detection (SURF – Speed Up Robust Feature, SIFT – Scale Invariant Feature Transform) and template matching were evaluated. Among all the technique, the most efficient and effective one – **context based template matching** technique is chosen as the implementation method to perform corresponding tasks.

The final deliverable of this project will benefits visually impaired in terms of bringing them happiness and confidence in personal and in the same time helps the visually impaired society and families to further reduce costs.

## **5.2 Improvement and Recommendation**

In order to have a better accuracy on detecting and tracking the user's finger, different value of skin colour should be used to create the skin filter (section 3.1.2.1) to achieve a maximum accuracy.

In order to further increase the accuracy of the finger detection module, higher weight should be allocated on the upper tips (the curve on the finger tip). By allocating more weight to the finger tip, objects such as orange coloured pencil will not be false detected as a finger.

A better quality of web camera will also increase the accuracy when the camera is able to provide features like autofocus, wide-range angle. Autofocus feature helps the system to obtain a sharper image and thus edges pixel can be extracted more in a more efficient and effective way.

Wide-range (or known as High Definition) angle camera will have a higher resolution. By changing the current camera with a higher resolution one, objects that are located further can be easily captured using wide-range angle camera, making the guidance process easier.

In order to hit the maximum performance of the object finder system, multicore programming on multicore processor should be implemented to take full advantage of the multichip processor. By implementing multicore programming, expensive calculation can be easily speeds up when the number of logical core processor increases.

## **5.3 Future Work**

### **5.3.1 – Object Detection**

Object detection module will be further developed due to the limited time and resources in this project. As object detection module part is already very complex module, the author simply assume that the targeted object is in single colour tone to ease detection.

After the completion of the project, more research will need to be carried out to focus on the object detection module to develop an intelligent solution on the module. Besides that, there is some possibility that the object detection module can comes from other researcher and be implemented into the system to produce a complete one.

### **5.3.2 – Further Improvement on the finger detection module**

In order to have a higher accuracy of finger detection, more research will need to be carried out in the future to improve on the current one. Technology such as Kinect by Microsoft can be implemented in the future to enhance the finger detection module. After such technology is implemented into the system, the system will be able to locate the depth of the object. Thus the navigation will no longer only limited to Left, Right, Up and Down, it will be enhanced to Left, Right, Up, Down, Front , and Back ( more to 3D ).



## References

Abdelsalam, Balaji & Steven, E., 2001. *Drishti. An Integrated Navigation System for Visually Impaired and Disabled*. [IEEE , 2001]

Garcia, C., 1999. Face detection using quantized skin color regions merging and wavelet packet analysis. [ACM, 1999]

Iwan, U. & Johann, B., 2001. *The GuideCane. Applying Mobile Robot Technologies to Assist the Visually Impaired*. [IEEE, 2001]

Julie, A.K., Shwetak, N.P. & Arwa, Z.T., 2006. *Where's My Stuff? Design and Evaluation of a Mobile System for Locating Lost items for the Visually Impaired*. [ACM,2006]

OpenCV Community, 2013. *Template Matching*. [Online] Available at: [http://docs.opencv.org/doc/tutorials/imgproc/histograms/template\\_matching/template\\_matching.html](http://docs.opencv.org/doc/tutorials/imgproc/histograms/template_matching/template_matching.html) [Accessed 3 April 2013].

Robin, L. & Arlene, G.R., April 2002. *Statistics on Vision Impairment. A Resource Manual*.

Truong, R.T.G.&.K.N., 2012. *CrossingGuard. Exploring Information Content in Navigation Aids for Visually Impaired Pedestrian*. [ACM, 2012]

Wee Ching, L. & K. H, M.L., 2007. *SIN. An Automated Navigation System for the Visually Impaired*.

# Object Finder For The Visually Impaired

by

**Mr. Lee Jia Hui**

A web cam software developed for the visually impaired to locate for a specific object.



For more information please contact :

Mr. Lee Jia Hui  
University Tunku Abdul Rahman  
Faculty of Information System And Technology (Perak)  
Bachelor of Information System (Hons) Information Systems Engineering

