INTELLIGENT LOCALISATION AND VISUALISATION SYSTEM FOR IN-PIPE INSPECTION PLATFORM

TAN WEI JUN

UNIVERSITI SAINS MALAYSIA

2017

INTELLIGENT LOCALISATION AND VISUALISATION SYSTEM FOR IN-PIPE INSPECTION PLATFORM

BY

TAN WEI JUN

Thesis submitted in partial fulfilment of the requirements for the degree of Bachelor of Engineering (Electronic Engineering)

June 2017

ACKNOWLEDGEMENT

I am grateful for having good health and mental well-being which has enabled me to complete this final year project. There are many people who deserve mentions and my sincere thanks for their help, guidance and support during the duration of this project. It is impossible for me to complete this dissertation without them.

First of all, I wish to express my sincere thanks to my final year project supervisor, Prof. Dr. Mohd Rizal Arshad, Dean of School of Electrical and Electronic Engineering USM for providing me with all the necessary guidelines and facilities to complete my project. Prof. Rizal has provided me with a clear concept on my project which leads to the success of my project. His patience in guiding me to come out with a good thesis is very much appreciated.

My sincere thanks too to Dr. Mohamad Adzhar Bin Md Zawawi, my academic advisor throughout the 4 years in my university life. Thanks for his expertise in providing me assistance throughout the project.

Many thanks to my friends, especially Khor Li Sing from the School of Manufacturing for helping me in many aspects along the way in this project. Because of her valuable suggestions from the robotic aspect, results of the project can be obtained successfully.

I am also thankful to my family and relatives for being supportive and providing endless encouragement in my studies. They fully understand the importance of this final year project and have urged and motivated me to do my very best.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CSI	Camera Serial Interface
CSS	Cascading Style Sheets
DOF	Degree of Freedom
FOV	Field of View
FPS	Frame per Second
GPIO	General Purpose Input Output
HTML	Hyper Text Markup Language
IoT	Internet of Things
IP	Internet Protocol
IMU	Inertia Measurement Unit
JSON	JavaScript Object Notation
LAN	Local Area Network
LED	Light-Emitting Diode

PCB	Printed Circuit Board
RPi	Raspberry Pi
RPM	Revolutions per Minutes
SSH	Secure Socket Shell

ABSTRAK

Pembetung adalah paip konkrit khas yang biasanya digunakan sebagai saluran salib untuk pembasmian parit dan laluan air di bawah jalan. Pemeriksaan untuk kakisan atau kerosakan pembetung adalah penting demi keselamatan. Robot merupakan teknologi masa ini yang boleh dipercayai dalam kerja pemeriksaan paip. Robot dapat meramalkan lokasi kawasan kerosakan dan mempunyai visualisasi masa nyata untuk mencapai keupayaan ini. Projek ini bertujuan untuk menyampaikan reka bentuk, pembangunan dan pelaksanaan sistem penyetempatan dan visualisasi untuk pemeriksaan pembetung. Dalam projek ini, satu sistem telah diperkenalkan untuk memperoleh dan memaparkan video masa nyata. Ia boleh digunakan untuk menentukan keadaan pembetung yang berdiameter besar (180mm-450mm). Sebuah kamera pemeriksaan bersama dengan lampu dipasang dan dilampirkan pada badan robot. IMU beserta dengan data telah dilampirkan. Pelayar web adalah platform untuk memperhati dan mengawal pergerakan robot.

ABSTRACT

Culvert is a special concrete pipe which is generally utilized as a cross channel for ditch alleviation and to pass water under the road. Inspection for corrosion or damage of the culvert is an important aspect of safety and using an in-pipe inspection robot is the most reliable and convenient way of doing this. It is necessary to obtain and predict the location of the damaged area and have a real time visualization to achieve this capability. This project is aimed at presenting the design, development and implementation of a localization and visualization system for culvert inspection platform. In this project, a system is introduced to acquire and display real-time video streaming so it can be used to determine the condition of large diameter culvert (180mm-450mm). An inspection camera together with a headlight are mounted on a swivel head attached to a body (robot) which has a wheel to enable it to move forward or backward. IMU module is attached to obtain several data. Web browser is the platform to visualize and control the movement of robot.

CHAPTER 1

INTRODUCTION

1.1 Background

Robotics is one of the fastest growing engineering fields of today. Robots are more and more being used for fully automated tasks which are very dangerous or when it would be unreasonable to ask a human to complete the task. For example, underground culverts or pipes that stretch across many miles. The inspection of pipes may be relevant for improving security and efficiency in industrial plants. All these specific operations such as inspection, maintenance, and cleaning are expensive, thus the application of the inspection robot appears to be one of the most attractive solutions.

It is decided that a robot to inspect the culvert has to be created with the goal of avoiding damage caused by cracks and corrosion of the culvert.

1.2 Problem Statement

The localization and visualization system must be able to use optical devices like camera to perform real-time video streaming.

The camera of the robot is expected to be on top or in front of the robot so that no obstacle would be there to block the view. Also, common techniques such as IMU sensor should be horizontally placed on the robot. Besides, dc motors has to be assembled with the robot chassis.

The system will need to have good lighting control system in order to perform the inspection as well as controlled lighting for visualization. So a suitable lighting control system has to be designed. Lastly, a webpage should be able to display particular functions such as real time visualization, orientation, location, and trajectory of the robot. It can also send commands to control the movement of the robot.

1.3 Objectives

The objectives of this project are:

1. To develop real time 3D visualization and localization system for culvert inspection platform.
2. To design a webpage to control the movement of the robot and display the real time visualization.
3. To calculate and estimate the trajectory of the movement of the robot.

1.4 Scope

The focus of this project is mainly on visualization and localization. Only 320x240 pixels of camera resolution is being used due to limited data bandwidth and computational power of the computer module. The results are obtained with real time processing instead of post processing in order to obtain and view the condition inside the culvert on time. Three axis of distance can be computed with no turning of motions involved.

1.5 Outlines of Report

This report consists of 5 Chapters. The first chapter is an introduction chapter which discussed the background of localization and visualization for in-pipe inspection system, problem statement, objectives of this project, scope and outlines of the report.

The second chapter is literature review. For this chapter, outcomes from researching on the scope of the project is covered to give an overview of what is available and the latest technology available in related to this project. Besides, theories relevant to this project are also included.

The third chapter describes the methodology for the development of localization for in-pipe inspection system. All steps involved are recorded and described in detail in this chapter. Also, all related theories and algorithms will be discussed.

In the fourth chapter, all the experiment results will be presented. The setup of the experiment will be described in detail with the help of attached photos. The data will be tabulated and the performance of the system will be analyzed and discussed.

The final chapter contains the conclusion regarding the project so as to conclude the achievement of the project's objectives. Suggestions for future research are also included in this chapter so that readers can have better idea of improvement for this project.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

There are some methods found that can visualize and localize the in-pipe inspection robot. Monocular vision camera can estimate the position and orientation of the robot by using monocular vision odometry technique. Before this technology was invented, wheel encoder algorithm was often used for positioning purpose. However, due to the wheel slippage, this algorithm is getting out of phase. It can be improved by using the combination of filtering encoder and inertia sensors. By fusing magnetometer and accelerometer data, Euler angles of the rigid body can be figured out. Lastly, web browser is becoming a widely-used platform for many software applications. It is possible to create a platform to visualize and control the robot.

2.2 Visual Odometry Based Technique

Visual odometry technique is where the captured images are used to estimate and determine the position and orientation of a mobile robot. In the use of an in-pipe inspection robot, it is assumed that the texture of the culvert is significant. The displacement vector between two positions can be found by comparing the current time image and previous time image. Generally, the displacement vector is 3 dimensional. It is represented by δx, δy, $\delta\theta$ as shown in figure 2.1.

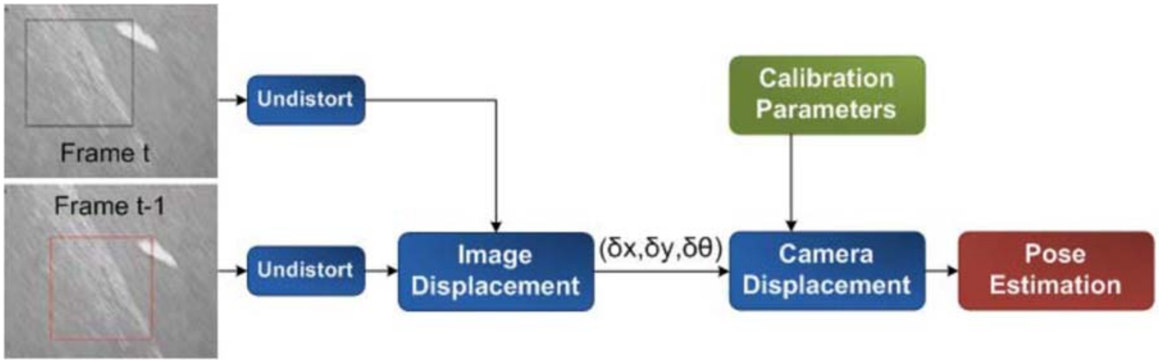

Figure 2. 1 Proposed monocular visual odometry algorithm [1]

The origin of the world coordinate system is located at the beginning and the center of the cylinder. The z- axis points is to the top and the y-axis points to the left. Also, the direction of the x-axis is defined using right hand rule. The position of the pipe inspection robot can be represented by a 3-dimensional vector (X_{world} , Y_{world} , θ) in which X_{world} and Y_{world} show the position of the robot in the world coordinate system and θ indicates the rotation of the robot around z-axis. [1]

2.3 Trajectory Estimation

The trajectory of the robot can be estimated by using visual odometry technique. Once the displacement vector is obtained, the trajectory of the robot can be computed by comparing the current image frame and previous image frame, by using the purposed the pixel displacement method.

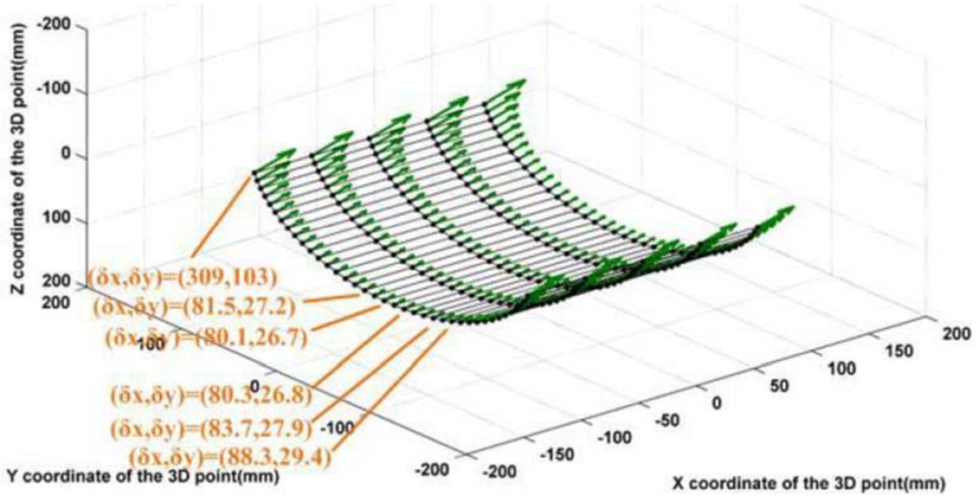

Figure 2. 2 Pixel displacement [2]

Figure 2.2 indicates the 3D derivative points for pixel displacement calculation. The total derivative can be calculated to obtain the pixel displacement vector. Assume the robot has moved 15 mm and 5 mm along its x-axis and y-axis respectively. This means that δX= 15, δY = 5 and δZ = 0. In Figure 2.2, 3D points as well as their pixel displacements are indicated. The field of view of the camera is 150° in this example. As in Figure 2.3, the pixel displacement in center of the image is smaller than the displacement in the sides of the image. It should be noted that there is a significant difference of displacement vector between the center of camera and the edge of camera. Therefore, the direct method of monocular visual odometry algorithm can be used to calculate the motion of the robot. However, if the field of view of the camera is reduced to 30 and its height is increased to 230 mm, the total derivative of 3D points changed to what is shown in the chart in Figure 2.4. [2]

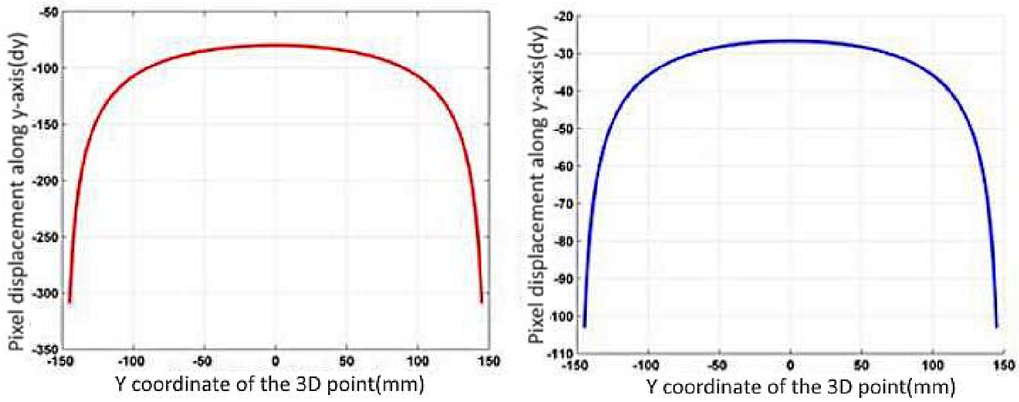

Figure 2. 3 Total derivative of 3D points for field of view of the camera [2]

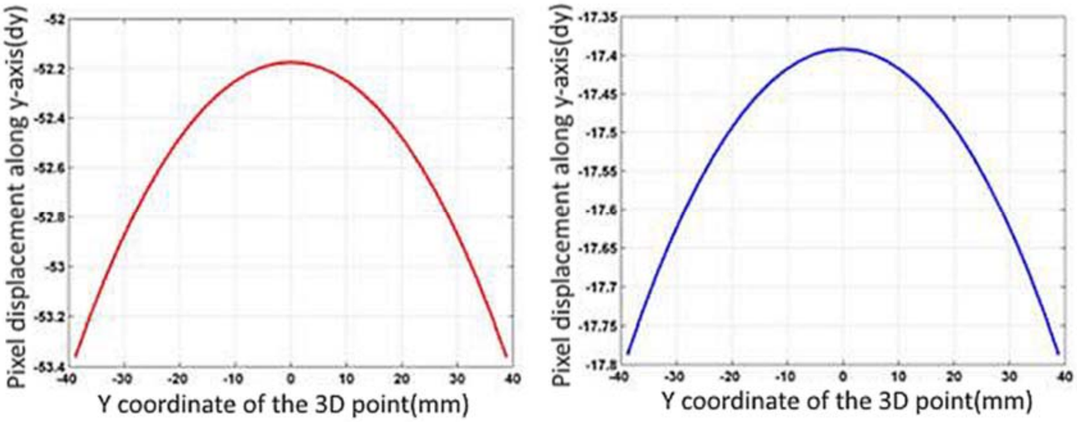

Figure 2. 4 Total derivative of 3D points for 30o field of view of the camera [2]

2.4 Localization Correction

The position estimation by wheel encoder often meets unpredictable error due to the slippage because of the variation of the surface's friction in different place. This problem affects the accuracy of the estimated robot's position. Therefore, the fusion of inertia sensors for mobile robot is able to determine more accurate and reliable position by using the Kalman's filter estimator.

The Kalman's filter consists of 4 processes, which includes prediction and estimation algorithm.

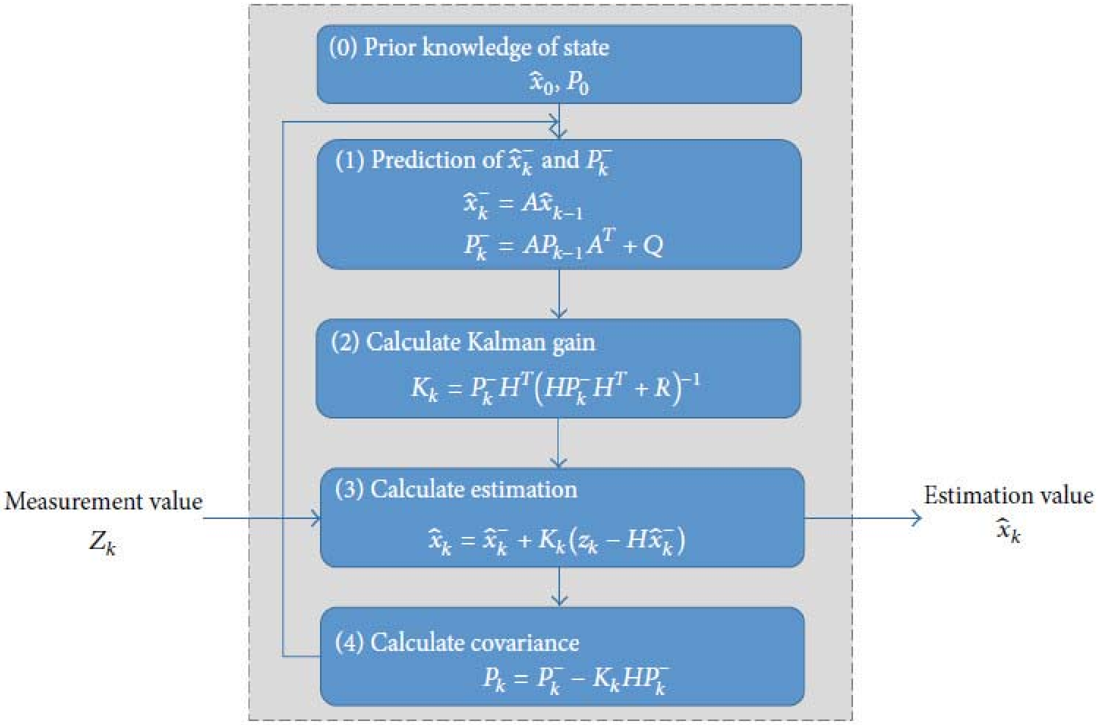
(0) Prior knowledge of state
$\hat{x}_0, P_0$
(1) Prediction of $\hat{x}_k^-$ and P_k^-
$\hat{x}_k^- = A\hat{x}_{k-1}$
$P_k^- = AP_{k-1}A^T + Q$
(2) Calculate Kalman gain
$K_k = P_k^- H^T(HP_k^- H^T + R)^{-1}$
Measurement value
Z_k
(3) Calculate estimation
$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$
Estimation value
$\hat{x}_k$
(4) Calculate covariance
$P_k = P_k^- - K_k HP_k^-$

Figure 2. 5 Kalman filter algorithm [3]

Starting from the first step as shown in figure 2.5 (1), the prior estimation value ($\hat{x}_{k-1}$) and covariance (P_{k-1}) is used to calculate the estimation value of $\hat{x}_k^-$ and P_k^-. These values are the estimation value, and will be used to calculate prediction for next step.

For prediction step, step (2), (3) and (4) are involved as shown in figure 2.5. The final value of $\hat{x}_k$ and P_k can be calculated by input the measurement value Z_k into. The covariance P_k is the criterion to determine the estimated value, x_k, which is the difference between actual value and the estimated value calculated by Kalman filter:

$x_k \sim N(\hat{x}_k, P_k)$

The x_k variable mean normal distribution that average value is $\hat{x}_k$ and covariance is Pk. Kalman filter algorithms choose estimation value using probability distribution of estimation value $x_k$which becomes the maximum probability value. [3]

2.5 Euler Angles Calculation

Accelerometer is very sensitive to the linear acceleration of 3 axes (x, y and z). It can measure the change in any linear acceleration according to the accelerometer's reference frame and earth's gravitational field vector. [5]

The orientation of a rigid body at a fixed frame can be described in many ways. One of the way is calculate the changes of orientation in yaw (ψ), pitch (θ) and roll (φ) using a three-axis accelerometer and magnetometer.

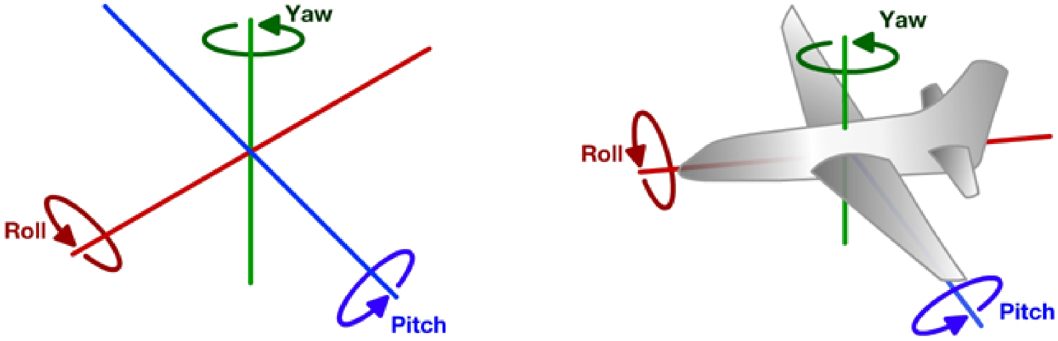

Figure 2. 6 Yaw-pitch-roll axes [18]

Figure 2.6 shows the model of the rotation of roll-pitch-yaw about the x, y and z axes respectively. With this model, assume the robot is in stationary and oriented in gravity, it produces the output of G_p given by:

$G_p = \begin{bmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{bmatrix} = Rg = R\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, where R is the rotation matrix and g is acceleration of gravity

Hence, the yaw, pitch and roll rotation matrices can be determined by:

$$R_x(\varphi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & -\cos\varphi \end{bmatrix} \qquad (1)$$

$$R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R_z(\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

From equation (1), we know that $G_p = R\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$, so the rotation matrix R can be computed as:

$$R\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = R_x(\varphi)R_y(\theta)R_z(\psi)\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \cos\psi\sin\theta\sin\varphi - \cos\varphi\sin\psi & \cos\varphi\cos\psi + \sin\theta\sin\varphi\sin\psi & \cos\theta\sin\varphi \\ \cos\varphi\cos\psi\sin\theta + \sin\varphi\sin\psi & \cos\varphi\sin\theta\sin\psi - \cos\psi\sin\varphi & \cos\theta\cos\varphi \end{bmatrix}\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \end{bmatrix} \qquad (2)$$

From (3), it is convenient to eliminate the yaw rotation angle ψ and so the roll φ and pitch θ angles can be calculated from R_{xyz} matrices. Equation (2) can be rewritten in the form of Equation (3) for further calculation of roll and pitch angles.

$$\frac{G_p}{\|G_p\|} = \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \end{bmatrix} = \frac{1}{\sqrt{{G_{px}}^2 + {G_{py}}^2 + {G_{pz}}^2}} \begin{bmatrix} G_{px} \\ G_{py} \\ G_{pz} \end{bmatrix} \qquad (3)$$

From Equation (3), the roll and pitch angles can be computed according to the rotation sequence Rxyz :

$$\tan\varphi = \left(\frac{G_{py}}{G_{pz}}\right)$$

$$\tan\theta = \left(\frac{-G_{px}}{G_{py}\sin\varphi + G_{pz}\cos\varphi}\right) = \frac{-G_{px}}{\sqrt{{G_{py}}^2 + {G_{pz}}^2}}$$

The yaw angle, ψ required an addition of magnetometer to determine. As a result, it can be calculated by:

$\psi = \mathrm{atan2}\big((-mY * \cos\varphi + mZ * \sin\varphi), (mX * \cos\theta + \mathrm{mY} * \sin\theta\sin\varphi * mZ * \sin\theta\cos\varphi)\big)$, where mX, mY and mZ are the three-axis reading from magnetometer and the atan2 [21] is arctangent with two argument.

$$\mathrm{atan2}(y, x) = \begin{cases} \arctan\frac{y}{x}, & if\ x < 0, \\ \arctan\frac{y}{x} + \pi, & if\ x < 0\ and\ y \geq 0, \\ \arctan\frac{y}{x} - \pi\,, & if\ x < 0\ and\ y < 0, \\ +\frac{\pi}{2}, & if\ x = 0\ and\ y > 0, \\ -\frac{\pi}{2}, & if\ x = 0\ and\ y < 0, \\ undefined, & if\ x = 0\ and\ y = 0 \end{cases}$$

2.6 Past, Present and Future of Robotic Tunnel Inspection

Nowadays, the majority of the culvert inspection is performed by the operator manually. It is dangerous to the operator as the environment conditions inside the culvert are full with dust, absence of light or presence of toxic substances. Robotic technology can replace the work of the operator and overcome these environmental issues. A few methods have been discussed.

Visual methods

Visualization is probably the most important feature for all types of pipe inspection. It can provide the actual visual information through capturing or streaming live video to the engineer. Visual information can be gathered in order to make initialization of the condition of the pipe inspection structure. It allows further investigations such as visual odometry technique or any other computer vision algorithm.

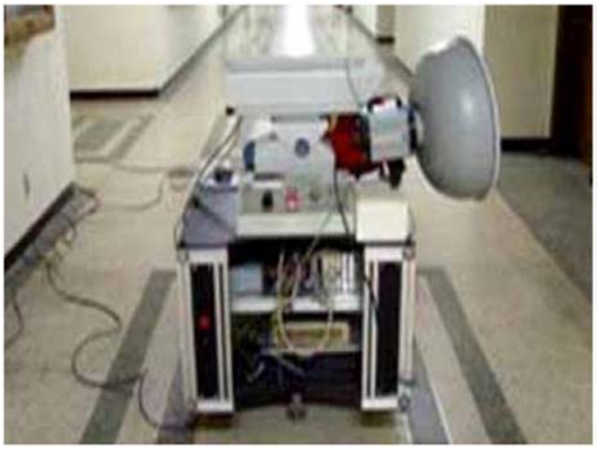

Figure 2. 7 Robotic platform with camera used in tunnel inspections [6]

Figure 2.7 shows the mobile robot equipped with a camera. The camera is mounted on the mobile robot with anti-vibration system to stabilize the vision system. The mobile robot goes through all the culvert and record the video while it is moving. Visual data (video) is collected and processed after the mobile robot back from culvert. Visual data is collected and then processed. In my opinion, this method is not efficient because it goes through 2 stages of process which are video recording and data processing. A real time visualization and processing method will bring further improvement.

Besides visual methods, there are a few more methods such as the strength based method, sonic and ultrasonic method, magnetic method, electrical method, thermography method, radar method, radiography method and endoscopy method discussed in the paper. However, these methods would not be considered in this project.

2.7 Trajectory Tracking Of Wheeled Mobile Robot

Trajectory tracking of kinematic modelling have been implemented and discussed.

Kinematic modelling studies the kinematic of motions without considering forces to be the affecting factors and it deals with geometric relationship of the mobile robot. For a mobile robot with the wheel’s radius of r placed from distance L/2 from the centre and θ as the robot orientation angle from x-axis, the equations are obtained as follow:

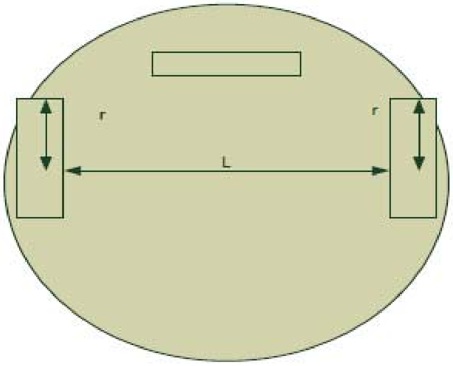

Figure 2. 8 Mobile robot scale [7]

$v = rw$

$w = \frac{d\theta}{dt}$, where w is the rotational speed.

The speed of each wheel is rw. Therefore, the linear velocity of the mobile robot is:

$v = \frac{r(w_L + w_R)}{2}$, which are w_L is the left wheel's rotational speed and w_R is the right wheel's rotational speed.

Consequently, the mobile robot's velocity can be represent in the form of:

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{r(w_L + w_R)}{2} \\ 0 \\ \frac{r(w_L - w_R)}{2} \end{bmatrix}$$

2.8 Web Browser as an Application Platform

Using a web browser as an application platform is becoming a hot topic since the widespread of the World Wide Web (www). It has always been possible to use programming languages such as Python, Java, PHP and others for creating websites to perform web-based application, but it was a rather tedious task. Web-based application means the applications live on the Web or Cloud as a service. It does not require installation of any specific software or update in order to use the application, but only web browser.

Almost all websites run on 3 components which are server, client, and database. In order to create a webpage to display real-time data, a dynamic website would be the choice. At a basic level, a dynamic website has the ability of updating and adding new content to the website. A dynamic website contains both client-side scripting and server-side scripting to generate the changing content.

The client can be any browser which the client is using to see a website, it can be laptop, mobile phone, or desktop. The server is at a specific location at anywhere to perform dedicated tasks like process data, acknowledging requests, and sending pages' source to the browser. In general, server-side scripting is executed by server whereas client-side scripting is executed by client's browser.

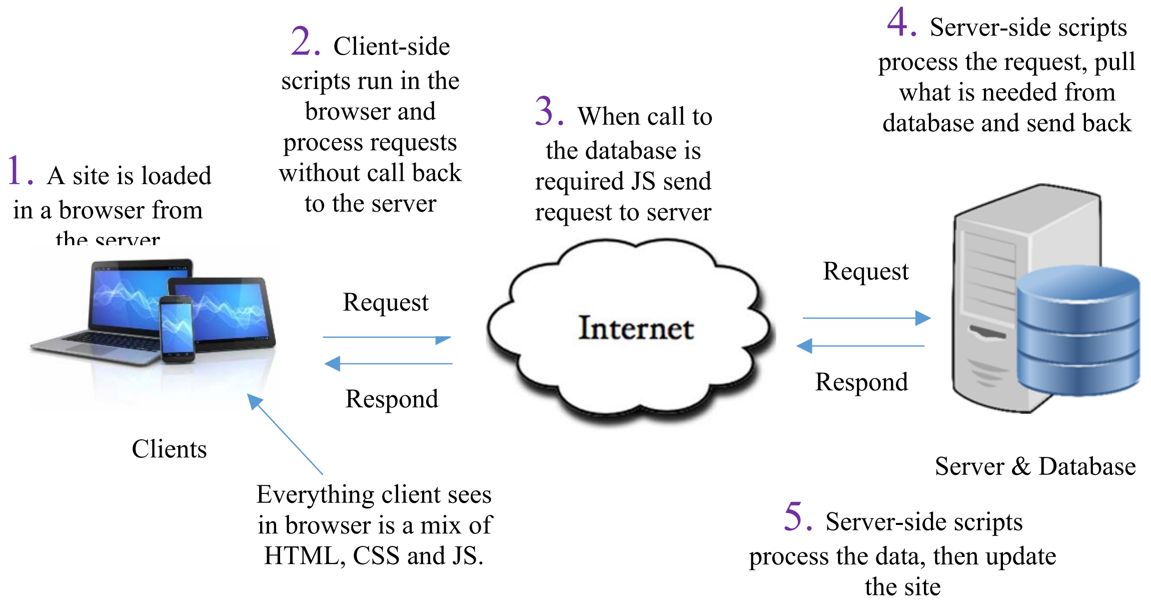
1. A site is loaded in a browser from the server
2. Client-side scripts run in the browser and process requests without call back to the server
3. When call to the database is required JS send request to server
4. Server-side scripts process the request, pull what is needed from database and send back
Request
Respond
Internet
Request
Respond
Clients
Server & Database
Everything client sees in browser is a mix of HTML, CSS and JS.
5. Server-side scripts process the data, then update the site

Figure 2. 9 The server, client and database work in tandem

2.9 Summary

There are some methods that can be used to estimate the position and orientation of the robot include monocular vision odometry method and fusion of inertia sensors with Kalman filter method. All these estimation methods are being implemented by researchers.

Since the camera is usually mounted with the mobile robot for culvert inspection, monocular vision odometry method is assumed to be the best method currently. However, there is a limitation which is the culvert pipe must have a large texture surface. However, for culvert inspection, it is assumed that the mobile robot inside the culvert is designed to move straight along the path and does not need to turn left or right. The depth is required to measure. Also, the robot is unnecessary to turn left or right, only forward and backward motions are needed to perform. Lastly, thanks to the web browser and World Wide Web, the visualization platform can be implemented.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

In this chapter, methods used to calculate the orientation of in-pipe inspection robot were explained in detail. Section 3.2 described the flow of the methodology used in this project. The project implementation flow was visualized with flowchart. Whilst section 3.3 discussed about the project requirements which included the hardware and software for this project. Section 3.4 was about data calculation and analysis, including camera resolution analysis. The method of calculating Euler angles from the data received from IMU was in this section. Besides that, the distance and position estimation were found in this section. Lastly, section 3.5 was about summarizing the methodology that has been discussed.

3.2 Project Implementation Flow

The research methodology is as shown in the flow chart below. Suitable hardware and software were chosen by considering the objectives to make the system works.

Software development and hardware design were implemented together so it can speed up the progress of work. Several calibrations such as IMU sensor and camera were made before running the required program script. If there was serious video stream delay on the real time visualization platform, software coding need to make a change to minimize the video stream delay as much as possible. If there was inaccurate or any undesired input from GY80 sensor, redesign of hardware part is needed to re-calibrate or reduce input disturbance.

Orientation in yaw (ψ), pitch (θ) and roll (φ) method would be used to estimate the robot's orientation. Python Flask library would be used to create web framework and launch local webpage for visualization purpose and as a platform to control the movement of robot.

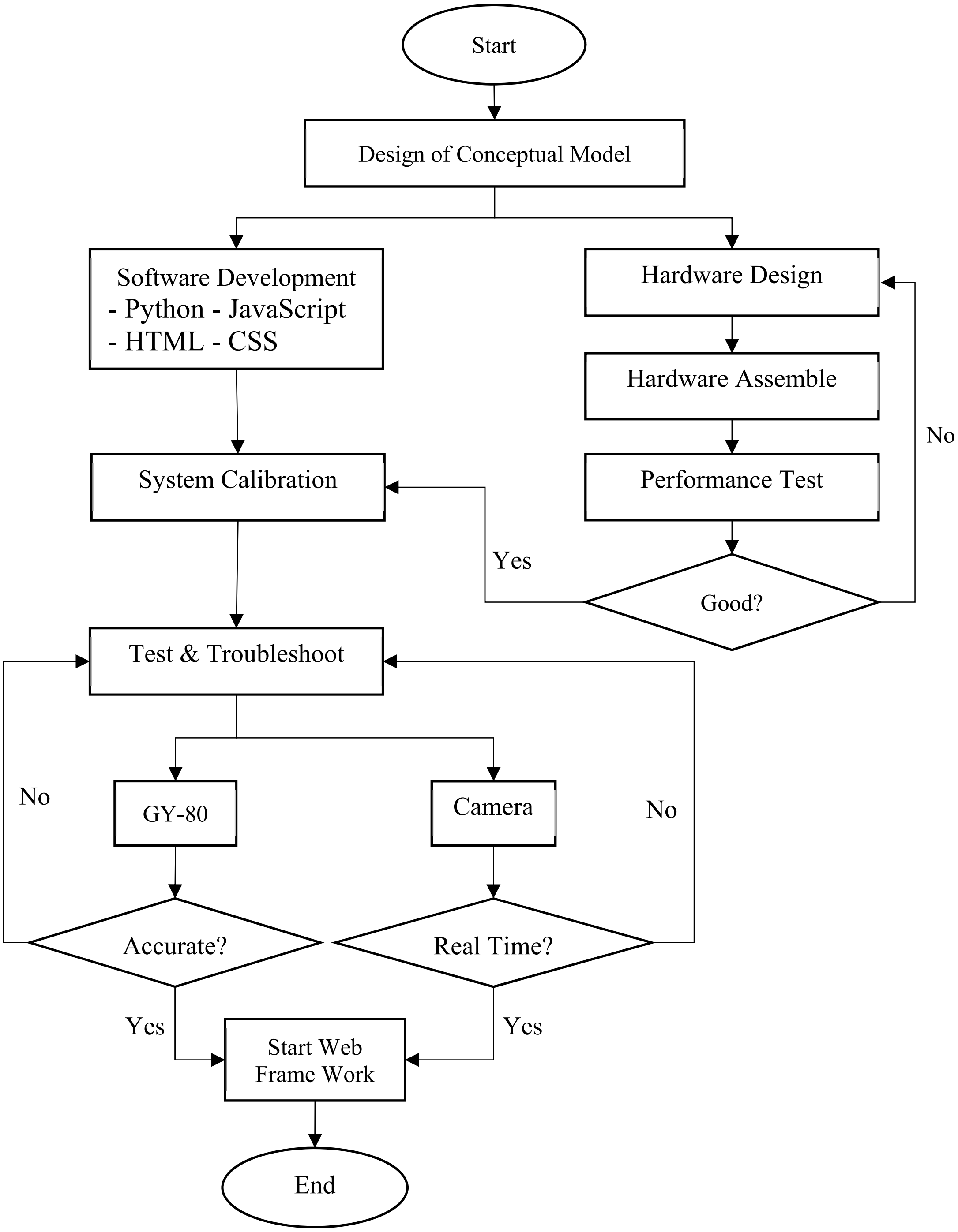

Figure 3. 1 Research methodology flow

3.3 Project Requirement

3.3.1 Hardware

The hardware description will be described and explained in more details in this section.

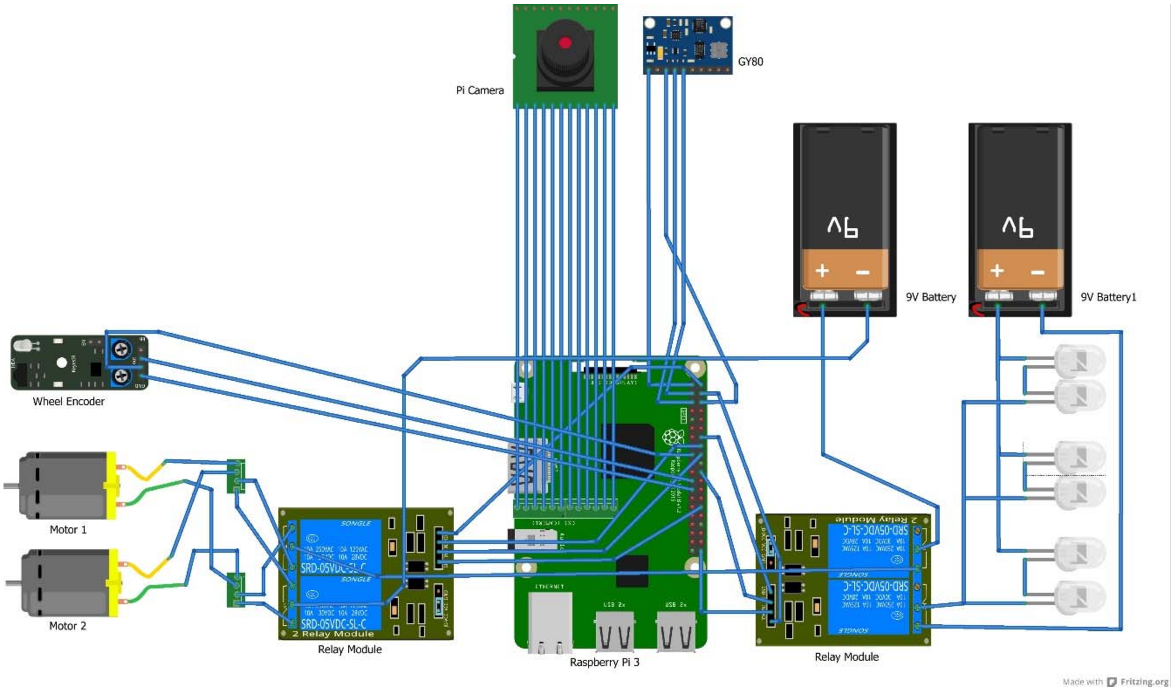

Figure 3. 2 Hardware connection diagram

The figure above shows the wire connection of the hardware part. A power bank was used as remote power source for RPi. However, dc motor consumed a lot of power in order to run it. This issue would cause insufficient current supply to the RPi and lead to malfunctioning of the RPi. So an external 9V battery is added as a power source to the dc motors. One of the 2-channel relay module was used as an electrical operated switch to control the power supply of the motors and LEDs. Another 2-channel relay module was used to control the direction of current flow whether the motor should rotate in clockwise or anti-clockwise direction. Besides, pi camera, GY-80 and wheel encoder sensor were directly connected to the RPi’s camera interface port and GPIO ports.

3.3.1.1 Raspberry Pi 3 Model B

Raspberry Pi 3 (RPi) was selected because we need a computer to interface with external devices and to set up a local web server, since microcomputer is more reliable compared to microcontroller. A microcomputer was released in 2016 and received acceptance across the board of employment among educators, developers and even engineers. With the best feature of its own, RPi 3 as shown in Figure 3.2 is featured with high efficiency core (ARM Cortex-A53 quad-core processor) which is equivalent to our old computer Intel Pentium IV processor. Only a little power supply is consumed in order to operate it due to its processor which applies RISC architectures similar to our smartphone nowadays. A 5 Volt and 1-2 Ampere power supply is already adequate for normal operation. Many other features are also offered such as Ethernet port, WiFi module, Bluetooth module, USB ports, GPIO pins and camera serial interface (CSI)[7]. With all these features, it fulfils the requirement for video streaming and sensor interfacing.

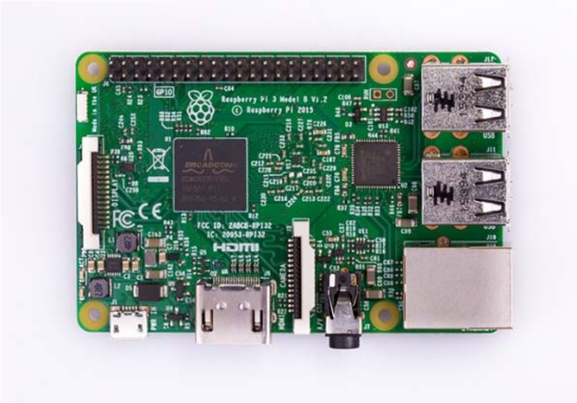

Figure 3. 3 Raspberry Pi 3 [8]

3.3.1.2 Raspberry Pi Camera Module

There are a lot of cameras which can interface and work with RPi, such as USB Webcam and RPi camera. RPi camera as shown in Figure 3.3 was chosen because:

- It is smaller compared to USB Webcam
- The PiCamera already has python library built-in in the Linux Raspbian OS
- A CSI-2 port was there to facilitate connection of RPi camera to RPi as shown in Figure 3.4.

Figure 3. 5 RPi camera module [10]

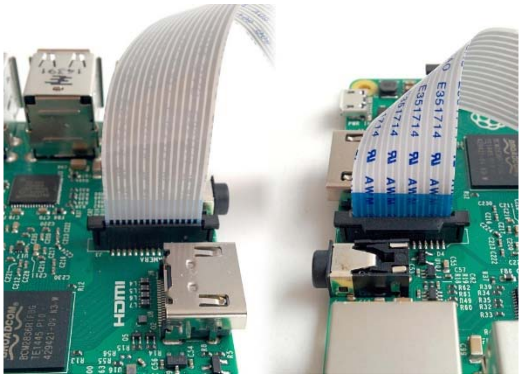

Figure 3. 4 CSI-2 Port [11]

3.3.1.3 GY-80 10 DOF IMU Module

GY-80 module was selected to track the robot position orientation instead of using visual odometry technique. The reasons were lesser programs need to be run but still enable to get accurate reading. It is a 10 DOF IMU module with these features:

- 3 Axis Accelerometer (ADXL345) [11]
- 3 Axis Gyroscope (L3G4200D) [12]
- 3 Axis Magnetometer (HMC5883L) [13]
- Barometer (BMP085) [14]
- Thermometer (BMP085) [14]

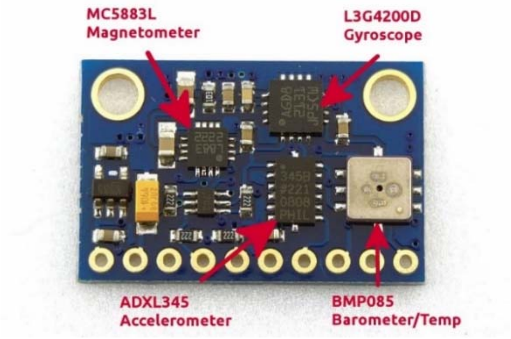

Figure 3. 6 GY-80 module [12]

3.3.1.4 Relay Module

A relay is an electrically operated switch. Most of the relays are used as an electromagnet to automatically control the switch and therefore provide electrical isolation of the current flow between 2 different circuits.

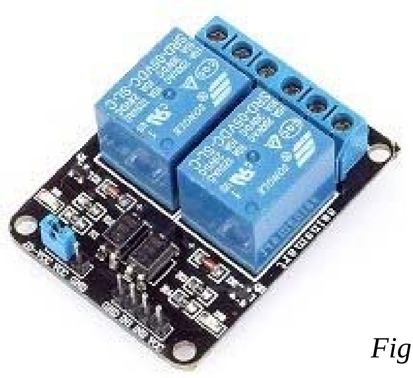

Figure 3. 7 2-Channel relay module [26]

3.3.1.5 Robot Car Chassis

A robot car chassis was obtained from Arduino robot car and designed to hold everything such as RPi, camera, IMU sensor, 9V batteries, wheel encoder, relay module and portable power bank as shown in Figure 10. It is very convenient for robot car project design and develop project plan because lesser work is required to assemble everything on it.

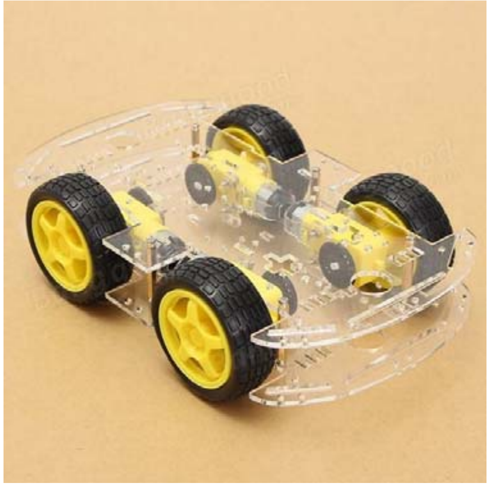

Figure 3. 8 Robot car chassis

3.3.2 Software

Software selection will be described and discussed in this section.

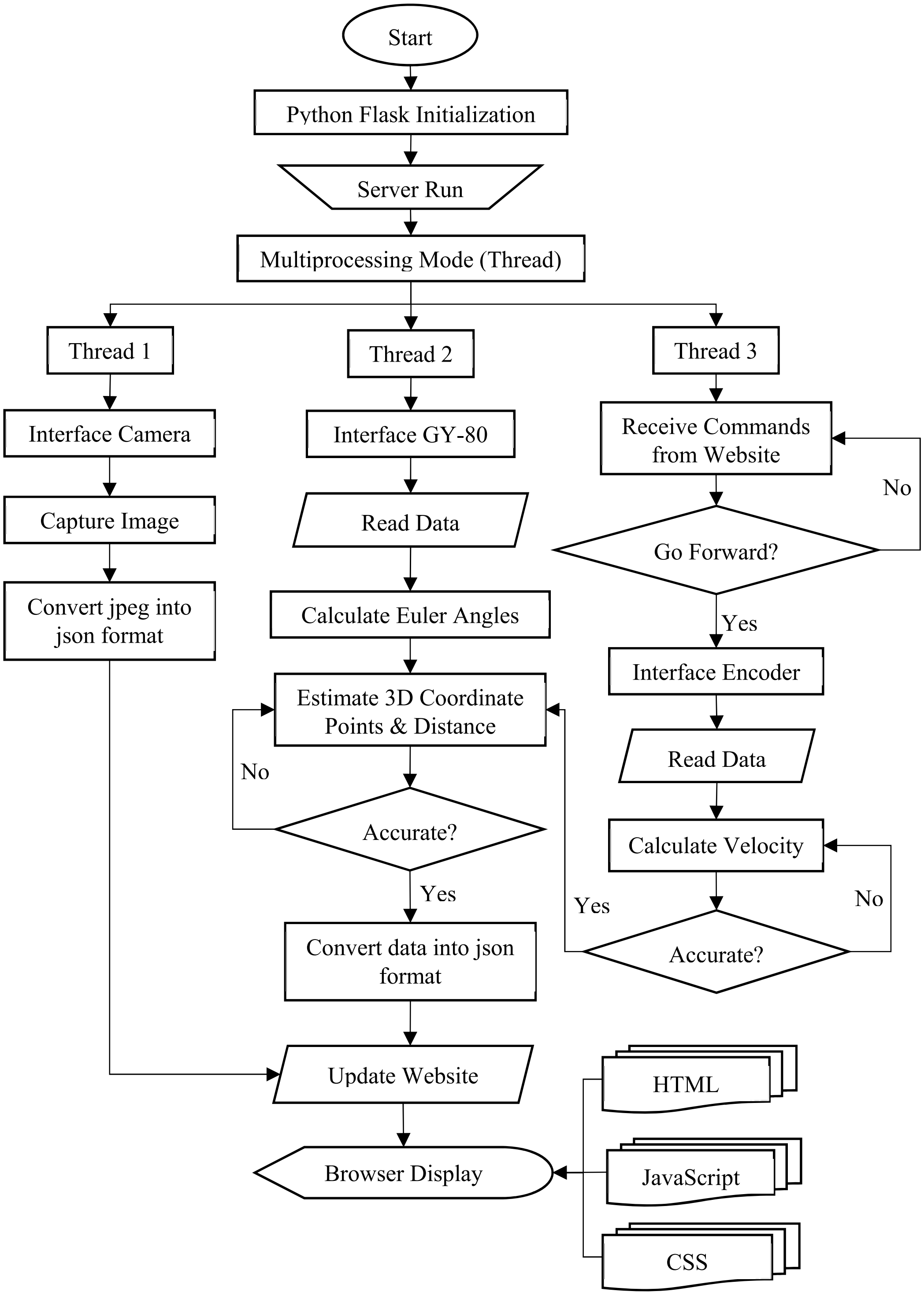

Figure 3. 9 Software implementation flow

The figure above shows the software implementation flow. Python Flask library was used to initialize the web framework and create a server web. Due to the real time visualization feature, the camera always captures a static image at frame rate of 20fps and keep replacing a new image every 0.05 seconds on website. With this method, a real time streaming video can be displayed on the browser.

Multiprocessing technique is applied because there are other sensors that need to be interfaced and data need to be calculated and analysed. Another two programmes were processed simultaneously together with the streaming video programme which are interfacing the IMU sensor (GY-80) and receive commands from website. Raw data from IMU sensor and wheel encoder are obtained in order to calculate the Euler angles and velocity of the robot when the motor is moving forward. Cartesian coordinates and distance travelled can be estimated through the raw data obtained and hence a 3D trajectory plot can be displayed on website.

3.3.2.1 Python Language

Python is a high level programming language which is widely used for general-purpose programming. It came from a same background with Perl languages [], which are from unix shell scripting but have different philosophy. One of the best advantage of using it is wide range of open source library support, including the RPi GPIO library to allow user interface external devices and flask library to create web framework.

Figure 3. 10 Logo of python language [17]