AUTONOMOUS TARGET TRACKING OF A QUADROTOR UAV USING MONOCULAR VISUAL-INERTIAL ODOMETRY

by

CHIN MUN XUE

Thesis submitted in fulfilment of the requirements for the
Bachelor Degree of Engineering (Honours) (Aerospace Engineering)

June 2018

ENDORSEMENT

I, Chin Mun Xue hereby declare that all corrections and comments made by the supervisor and examiner have been taken consideration and rectified accordingly.

____

(Signature of Student)

Date:

____

(Signature of Supervisor)

Name:
Date:

____

(Signature of Examiner)

Name:
Date:

DECLARATION

This thesis is the result of my own investigation, except where otherwise stated and has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any other degree.

(Signature of Student)

Date:

ACKNOWLEDGEMENTS

“We often take for granted the very things that most deserve our gratitude.”

~ Cynthia Ozick

To accomplish success in life, we have to cease dwelling on the negative experience in the past and quit limiting our potential by constantly thinking about what we have done poorly. Easier said than done, this is definitely the biggest leap ever over the lava of fear for failure. Looking back when I first embark on the journey of my final year project, I am without any clear directions. In other words, I am lost; the thought “I can’t do it” has always played with my mind. Yet, writing this at this moment, I have fought hard and won the battle, kind of. I would like to share my happiness with and express my gratitude to a few important people standing by me side-by-side along the fight; here is to victory!

“It takes a big heart to help shape little minds.”

~ Author Unknown

Foremost, I have to and must convey my sincere appreciation to my supervisor, Dr. Ho Hann Woei. He is a very knowledgeable man in the subject matter of this thesis yet I have almost zero understanding towards it. Hence, I believe that it takes a great lot of patience to guide me to the right direction so that I may get a little grasp on the basic and fundamental principles. He would also try his best to allocate time to have meetings frequently, listen to my problems actively and give constructive suggestions, despite being busy with his research work and another final year student who is also under his supervision. Though it may seem to be trivial, I feel indeed grateful to this.

"Call it a clan, call it a network, call it a tribe or call it a family. Whatever we call it, whoever we are, we need one."

~ Jane Howard

My parents and siblings are always be there for me; they are my pillars of strength. My parents want me to be happy and righteous over successful. They would tell me it is okay not to gain success but have to learn from the failure and do the right things only no matter what. Of course, they would be even gladder if I can have the accomplishment. My siblings act as a listener, guider and supporter, at the same time. Whenever I have encountered obstacle in my way and become stuck, they are very willing to lend their ears and give me encouragement and moral support. It would not be over to say, without them, I shall never come this far. I am truly indebted.

Last but not the least, I would like to seize also this opportunity to say thank you to my friends who have been my companions for the past four years, making this university life of mine a meaningful and memorable one. To sum it up, it is definitely a blessing to have my supervisor, parents and siblings as well as friends. Many thanks to myself for making it all the way out with all the helping hands. I hope God may bless us in our future endeavours. Till next time!

AUTONOMOUS TARGET TRACKING OF A QUADROTOR UAV USING MONOCULAR VISUAL-INERTIAL ODOMETRY

ABSTRACT

Unmanned Aerial Vehicle (UAV) has been finding its ways into different applications. Hence, recent years witness extensive research towards achieving higher autonomy in UAV. Computer Vision (CV) algorithms replace Global Navigation Satellite System (GNSS), which is not reliable when the weather is bad, inside buildings or at secluded areas in performing real-time pose estimation. The controller later uses the pose to navigate the UAV. This project presents a simulation of UAV, in MATLAB & SIMULINK, capable of autonomously detecting and tracking a designed visual marker. Referring to and improving the state-of-the-art CV algorithms, there is a newly formulated approach to detect the designed visual marker. The combination of data from the monocular camera with that from Inertial Measurement Unit (IMU) and sonar sensor enables the pose estimation of the UAV relative to the designed visual marker. A Proportional-Integral-Derivative (PID) controller later uses the pose of the UAV to navigate itself to be always following the target of interest.

PENJEJAKAN SASARAN SECARA AUTOMATIK OLEH PESAWAT UDARA TANPA JURUTERBANG MENGGUNAKAN ODOMETER PENGLIHATAN MONOKULAR DAN UNIT PENGUKURAN INERSIA

ABSTRAK

Pesawat Udara Tanpa Juruterbang telah digunakan dalam pelbagai aplikasi. Oleh itu, banyak penyelidikan telah dijalankan untuk mencapai autonomi yang lebih tinggi dalam penggunaan Pesawat Udara Tanpa Juruterbang. Algoritma Visi Komputer telah menggantikan Sistem Satelit untuk Navigasi Global yang tidak begitu menyakinkan, terutamanya ketika cuaca buruk, di dalam kawasan bangunan atau di kawasan terpencil dalam penganggaran kedudukan dan orientasi masa sebenar Pesawat Udara Tanpa Pemandu. Sebuah pengawal akan menggunakan anggaran kedudukan dan orientasi masa sebenar tersebut untuk mengemudikan Pesawat Udara Tanpa Juruterbang. Projek ini mempersembahkan sebuah simulasi tentang kemampuan Pesawat Udara Tanpa Juruterbang dalam pengesanan dan penjejakan sebuah penanda visual secara automatik. Sebuah pendekatan baru telah dirumuskan untuk mengesankan penanda visual setelah merujuk kepada algoritma Visi Komputer yang sedia ada. Penggabungan informasi daripada algoritma Visi Komputer, Unit Pengukuran Inersia dan penderia sonar membolehkan perlakuan penganggaran kedudukan dan orientasi Pesawat Udara Tanpa Juruterbang relatif kepada penanda visual. Kemudian, sebuah pengawal PID akan menggunakan anggaran kedudukan dan orientasi itu untuk mengemudikan Pesawat Udara Tanpa Juruterbang supaya sentiasa mengikuti sasaran tersebut.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CV	: Computer Vision
D	: Dimensional
DOF	: Degree-of-Freedom
EKF	: Extended Kalman Filter
FOV	: Field of View
GNSS	: Global Navigation Satellite System
GPS	: Global Positioning System
IBVS	: Image Based Visual Servoing
PBVS	: Position Based Visual Servoing
PID	: Proportional-Integral-Derivative
RGB	: Red-Green-Blue
RMS	: Root Mean Square
ROI	: Region of Interest
RTK	: Real Time Kinematic
SIFT	: Scale Invariant Feature Transform
UAV	: Unmanned Aerial Vehicle
VRML	: World Reality Modelling Language

LIST OF SYMBOL

a	: Major Axis Length
b	: Minor Axis Length
$D_{centroids}$	: Distance between Two Image Regions Centroids
d	: Diameter of Two Adjacent Image Regions
m	: Row
n	: Column
$Re_{x,y}$	: x, y-axes Image Resolution
r	: Radius of Two Adjacent Image Regions
X, Y_{actual}	: x, y-axes World Coordinates
$X, Y_{relative}$	: Distance between the UAV and the Detected Visual Marker
x, y_{image}	: x, y-axes Image Coordinates
η	: Diameter Ratio of Two Adjacent Image Regions
θ, φ	: Pitch, Roll Angles

CHAPTER 1

INTRODUCTION

1.1 Unmanned Aerial Vehicle (UAV) and Its Application

Unmanned Aerial Vehicle (UAV), as its name suggests, is a space-traversing vehicle, which flies without a human crew on board, either controlled remotely or operated autonomously. There are various categories of UAV: single rotor, multi rotor, fixed wing and hybrid, shown in Figure 1.1.

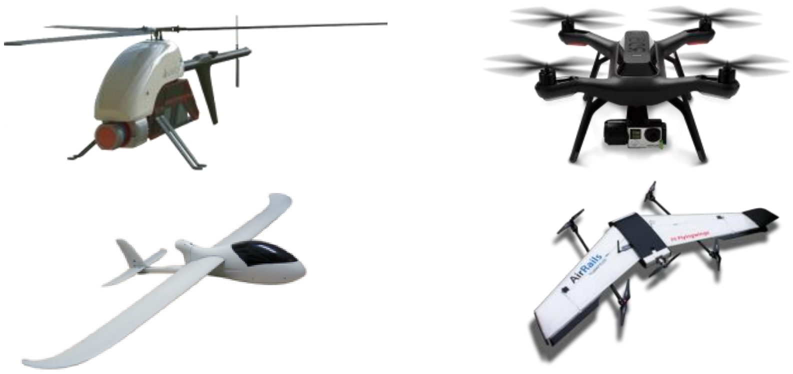


Figure 1.1: The top left is single rotor, top right is multi-rotor, bottom left is fixed-wing and the bottom right is hybrid UAV, respectively.

Distinguished by attractive features like low cost, high maneuverability and easy maintenance, UAV has found its way into different applications. Hobbyists usually control UAV remotely to take aerial photograph or video while UAV, which operates autonomously, has wider uses in military and security, photogrammetry and remote sensing, surveillance and safety, forestry and agriculture, urban planning, architecture, civil engineering and mining, energy and power industry as well as delivery (Arfaoui, 2017). This has significantly reduced the unnecessary workload or even risk taken by humans to perform the tasks listed above. For example, when a tsunami struck the Fukushima nuclear plant in Japan on 11 March 2011 and caused leakage of radioactive substances, it is important to identify the severity of contamination before taking further measure to tackle the problem. However, exposing to radiation is hazardous or even fatal to humans. To cope with this issue, Japanese company has developed and deployed a drone (i.e. TEPCO) to measure the radiation levels inside the reactor buildings.

Given the vast advantage and potential in making use of autonomous UAV to accomplish tasks, the last decade sees extensive research towards developing UAV with higher autonomy. The research mainly focuses on four areas: (1) platform design and fabrication, (2) dynamic modelling, (3) navigation and guidance algorithms design and implementation as well as (4) flight control (Cai et al., 2014). This thesis focuses more on the third area. To enhance the understanding of the readers, it is very vital to give a clear definition to the term navigation and guidance, respectively.

Navigation generally means the monitoring and controlling of movement, includes the process of acquiring, analyzing data as well as extracting and inferencing information about a vehicle's states and its surrounding environment in order to safely and successfully accomplish the mission objective (Kendoul, 2012).

A guidance system is the "driver" of the vehicle, which makes decision to achieve the assigned mission or goal. In other words, a guidance systems functions to replace the cognitive processes of a human pilot or operator. To make the decision, it requires information from the navigation system (Kendoul, 2012).

1.2 Navigation Problems of UAV

One of the most challenging obstacles in the third aspect aforementioned is the state estimation of UAV. This mainly concentrates on the processing of raw sensor measurements to estimate the state variables, especially those related to real-time pose (i.e. position and orientation). This is largely due to the constraints placed on the weight of the sensors bearable by the UAV, the power required by the onboard sensors and cost. Onboard sensors with high processing power are usually more power consuming and expensive.

1.3 Global Navigation Satellite System (GNSS)-Aided Inertial Navigation

To date, a large number of autonomous navigation systems available employ the combination of the Inertial Measurement Unit (IMU) and a Global Navigation Satellite System (GNSS), usually the well-known Global Positioning System (GPS). Even though the method mentioned above works well for most environments in clear weather, it is not a fully reliable system. This is due the fact that the GNSS, which is an external signal, can be lost for several reasons: during cloudy weather, inside any buildings or at secluded areas. In such cases, the UAV cannot function properly and may even incur an accident. Hence, in order to enable UAV to work in a GPS-degraded/ denied environment, there is a paradigm shift to computer vision (CV) algorithms, especially the use of monocular camera to receive information about the surroundings, in recent years. This is because monocular camera has less weight and is more cost-effective than having stereo camera. Therefore, monocular camera is more preferable.

1.4 Visual-Aided Inertial Navigation

There are feasible CV algorithms such as Feature Points and Template Matching with or without visual marker, either natural or artificial. For example, Scale Invariant Feature Transform (SIFT) and correlation coefficient-based method is one of the examples for Features Points and Template Matching, respectively. The markers normally have some recognizable and unique features so that analysis and extraction of information are easy to perform. Various artificial visual markers are available also such as helipad-shaped (Patruno et al., 2017) (Santoso et al., 2016), concentric-circle-based (Nguyen et al., 2017) (Cucci, 2016) and squared markers (Babinec et al., 2014), as shown in Figure 1.2. Sometimes, landmarks are also the target of interest. Figure 1.3 shows a few examples.

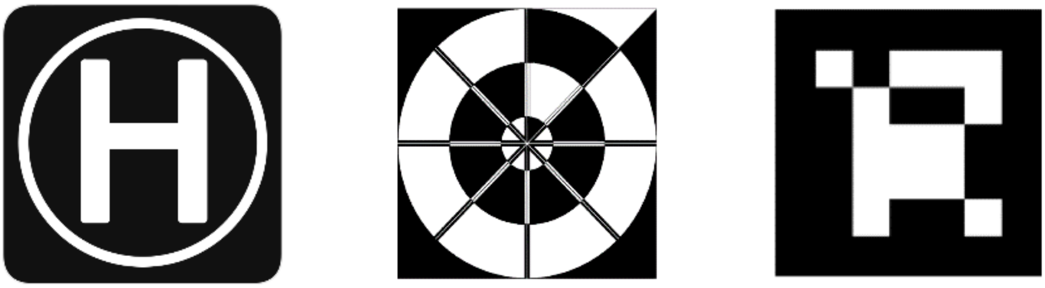


Figure 1.2: At the left is helipad-shaped; in the middle is concentric-circle-based and at the right is squared marker, respectively.

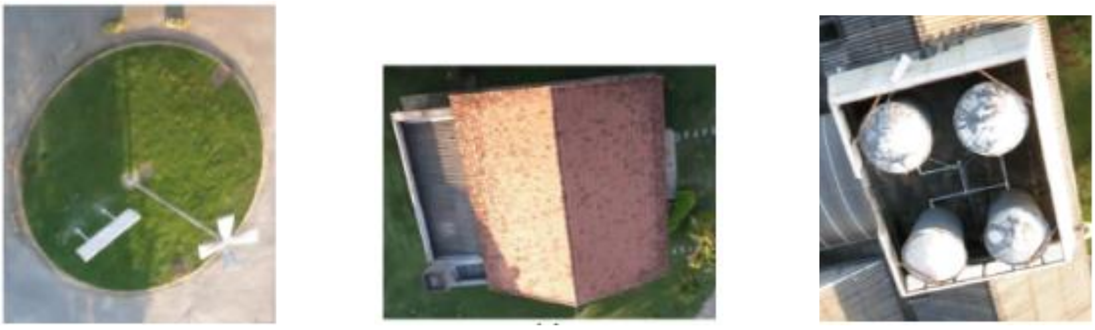


Figure 1.3: At the left is a roundabout; in the middle is the roof of a house and at the right is the roof, respectively.

Generally, a CV algorithm is in the following sequence: (1) image acquisition, (2), image preprocessing as well as (3) extracting and labeling information. Either a monocular or a stereovision camera can do image acquisition. After acquiring the image, it undergoes filtering, segmentation and thresholding. Filtering serves to add soft blur, sharpen details, accentuate edges or remove noise, segmentation finds groups of pixels, which "go together" while thresholding removes low-strength contour segments by setting the pixels that have intensity values above a threshold to a foreground value and other remaining pixels to a background value. Later, the information (i.e. coordinates) is extracted and labeled.

A CV algorithm gives either the two dimensional (2D) features or the three dimensional (3D) pose measurements, and together with the data from IMU, these are useful information in the closed loop control system. The process is visual servoing; and, in general, there are a few categories: Image Based Visual Servoing (IBVS); Position Based Visual Servoing (PBVS) and hybrid visual servoing. Figure 1.4 illustrates the control structure for the first two groups while Table 1.1 summarizes the advantage and disadvantage for both techniques (Szeliski, 2010).

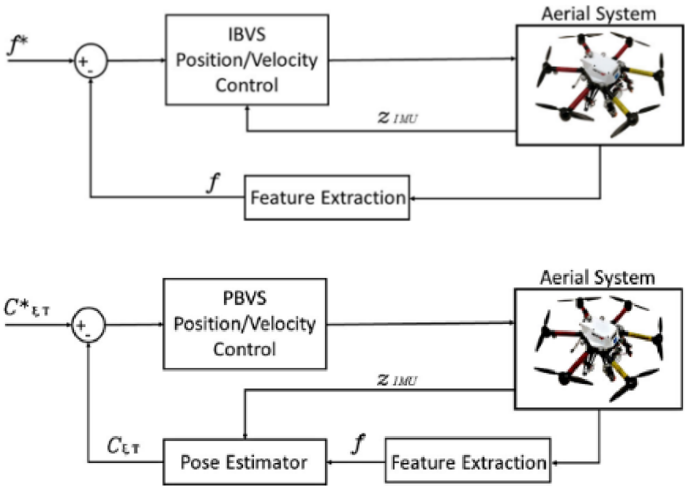


Figure 1.4: The top is the IBVS while the bottom is the PBVS control structure.

Table 1.1: It shows the advantage and disadvantage of IBVS and PBVS.

	Advantage	Disadvantage
IBVS	- When specified the desired robot pose, it is robust to calibration and model errors.	- The induced 3D trajectory is not predictable and maybe unsatisfactory.
PBVS	- Globally asymptotically stable when pose assumes perfect estimation.	- Additional information (i.e. true camera parameters) to estimate the pose of camera.

Stated in Table 1.1, PBVS needs extra information to estimate the pose of camera, in this case, mounted on UAV. This can be done by fusing information obtained (i.e. 2D position of the image captured) from the CV algorithm with the linear acceleration and angular rate given by IMU. Since the camera does not provide the depth (i.e. the vertical distance, in the z-axis direction, from the UAV to the target) of the target, a barometer or sonar sensor is useful in measuring the altitude of the UAV with respect to the visual marker. Although visual-aided inertial navigation systems see bright prospect in achieving higher autonomy in UAV flight, it is yet to have a CV algorithm providing stable results.

1.5 PID Controller of UAV

A controller later uses the estimated pose of the UAV to perform navigation. Some of the common controllers are PID, Fuzzy or Fuzzy PID controller. PID controller is renowned for its robust performance in a wide range of operating conditions and functional simplicity. Let probe deeper into the popular PID controller.

Below includes the discussion for each basic coefficient of PID controller. The proportional part collects sensor data in the present, integral part in the past and the derivative part in the future. The basic idea behind a PID controller is to calculate the difference between a desired set point or reference value (i.e. error value) and a measured process variable as well as applies a correction based on the three terms. The controller attempts to minimize the error over time.

1.6 Problem Statement and Research Questions

Referring to the challenges and problems mentioned above, this thesis has the following main problem statement: how UAV can utilize monocular visual-inertial odometry to track a target of interest (i.e. the designed visual marker) autonomously?

To detect a target of interest, it is inevitable detect and recognize it in the first place. Therefore, the problem statement should actually comprises two parts: (1) detection and (2) tracking.

The detection part acquires images with the designed visual marker using a monocular camera and later identifies and recognizes it using a CV algorithm. This leads to the first research question: how UAV can utilize a CV algorithm to distinguish the target of interest from an image or video frame?

The tracking part follows the designed visual marker using a combination of information from the monocular camera and the IMU. This formulates the second research question: how UAV can utilize a combination of information from the monocular camera and the IMU to trace the target of interest from an image or video frame?

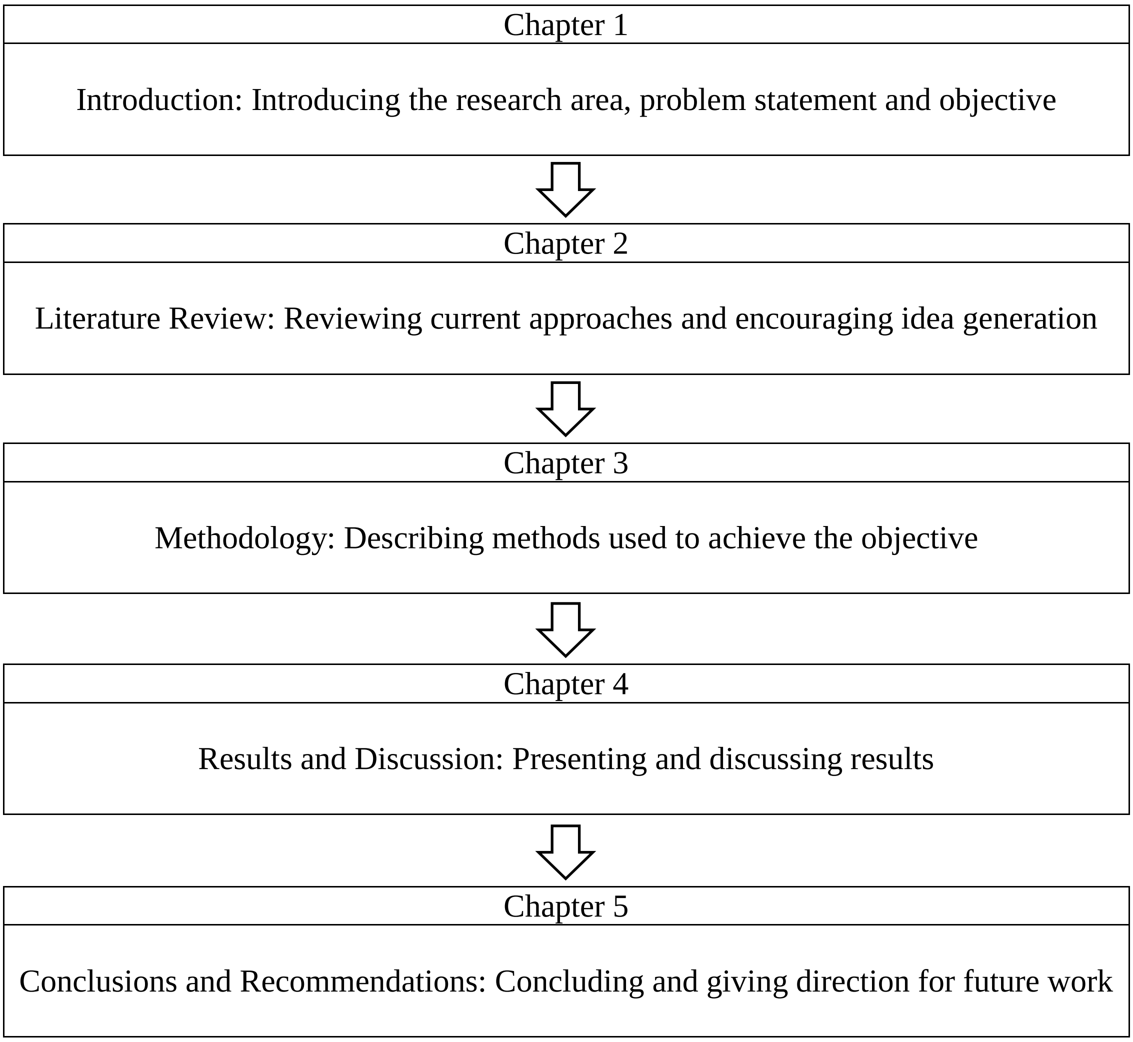
1.7 Research Objectives

To answer the research questions, setting of the following research objectives is important. It helps to focus on achieving the goals of this final year project and minimizing the possibility of a digression. There are five research objectives:

- To design a visual marker which has to be unique and recognizable
- To provide classical Blob Detection to detect the marker
- To use the IMU data to compensate for the camera rotation
- To simulate the UAV tracking the marker using a PID controller

1.8 Outline of the Thesis

This outline of the thesis is as shown in Figure 1.5. Chapter 2 focuses on reviewing current approaches by previous researchers on relevant or similar area. This is to avoid from reinventing the wheel and at the same time encouraging idea generation. Chapter 3 describes methodology for detecting the designed visual marker in terms of MATLAB coding, tracking the designed visual marker, and visualizing the simulation of the UAV to detect and track the marker. Chapter 4 presents the visual results and discussion. Chapter 5 summarizes and concludes the results obtained and gives recommendations for future work.



Chapter 1
Introduction: Introducing the research area, problem statement and objective
Chapter 2
Literature Review: Reviewing current approaches and encouraging idea generation
Chapter 3
Methodology: Describing methods used to achieve the objective
Chapter 4
Results and Discussion: Presenting and discussing results
Chapter 5
Conclusions and Recommendations: Concluding and giving direction for future work

Figure 1.5: It shows the outline of the thesis.

CHAPTER 2

LITERATURE REVIEW

2.1 Global Navigation Satellite System (GNSS)-Aided Inertial Navigation

Mentioned in Chapter 1, there are several solutions in estimating the state of a craft or vehicle. One of them is the GNSS-aided inertial navigation.

Integrating Global Position System (GPS) and Inertial Measurement Unit (IMU) can estimate the full kinematic state of a vehicle, along with sensor error parameters. There are twenty-two state variables identified: sixteen describes the full motion of the craft in 6Degree-of-Freedom (6DOF) while the other six describes the parameters of sensors error. Since GPS provides information at a low rate while IMU gives high frequency information, incorporating Extended Kalman Filter (EKF) helps to fuse the information of different frequencies. This approach is presented in (Abdi et al., 2016).

To increase the robustness of GNSS-aided inertial navigation in navigating a UAV, using Real Time Kinematic (RTK) GPS is one of the options (Gryte et al., 2017). It is a complicated technique but the basic concept is to reduce and remove errors common to a base station and rover pair, as shown in Figure 2.1.

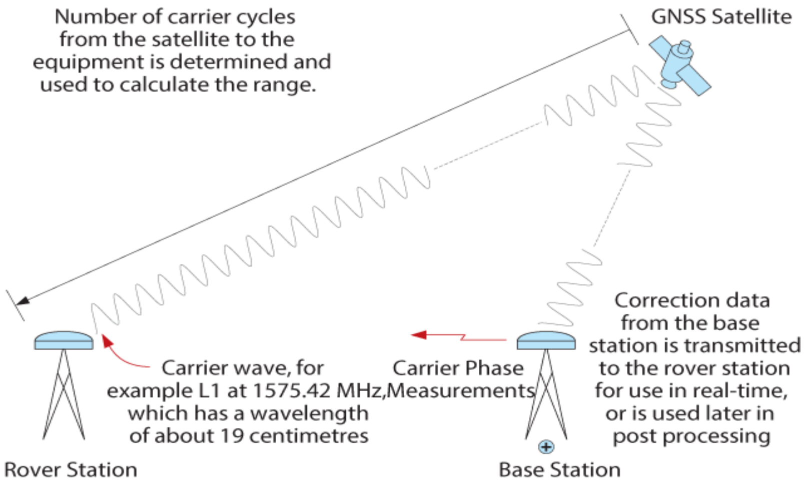

Number of carrier cycles from the satellite to the equipment is determined and used to calculate the range.
GNSS Satellite
Correction data from the base station is transmitted to the rover station for use in real-time, or is used later in post processing
Carrier wave, for example L1 at 1575.42 MHz, which has a wavelength of about 19 centimetres
Carrier Phase Measurements
Rover Station
Base Station

Figure 2.1: It shows the concept of RTK GPS.

Although it may seem feasible for GNSS-aided inertial navigation system to work, it is not without its disadvantages. For GPS, the signals become weak when the sky is cloudy or absent when inside buildings and at secluded areas. For RTK GPS, the referenced document itself states that the drawbacks are the sensitivity to agile maneuvers and obstructed GNSS signal path. Furthermore, it requires an additional rover pair, which may cause extra cost.

2.2 Visual-Aided Inertial Navigation

In fact, there are fervent research on visual-aided inertial navigation system. As mentioned in Chapter 1, there are computer vision (CV) algorithms using Feature Points and Template Matching to detect artificial visual markers or landmark to estimate pose. One of the popular artificial visual markers is the standard helipad (i.e. made of the marker H inside circles); it locates at airports, on top of skyscrapers, on large ships and oil platforms (Patruno et al., 2017). There are numerous algorithms to detect a helipad and one of those is corner detection and matching. It includes a few steps: (1) image acquisition and preprocessing, (2) helipad mark extraction, (3) corner detection of helipad, (4) corner matching and (5) pose estimation. Figure 2.2, 2.3 and 2.4 shows (1), (2) and (4), respectively.

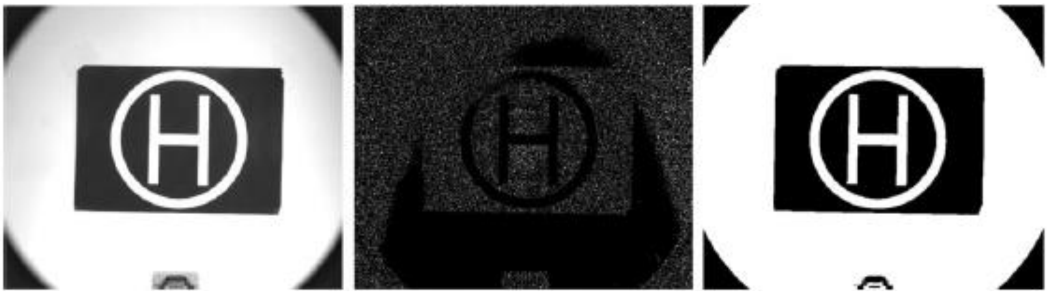


Figure 2.2: The left hand side is the output image after applying the Wiener filter, the middle is the noise detected after the application of median filtering and the right hand side is the resulting binary image.

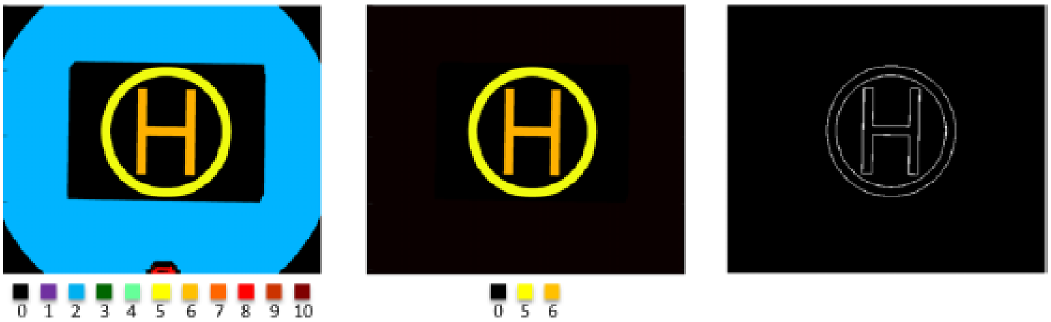


Figure 2.3: The left hand side indicates all the possible candidates identified before the cascade check, the middle is the final helipad marks represented by the two remaining blobs and the right hand side is the contours of helipad.

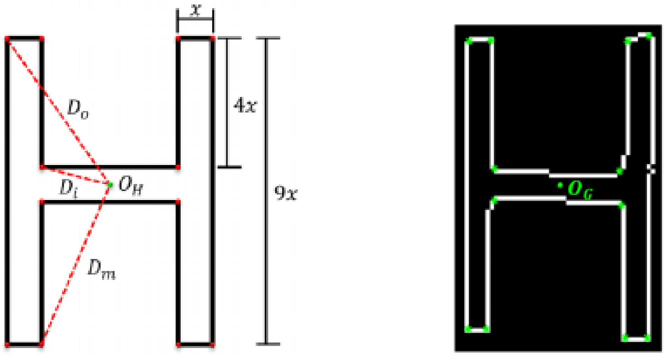


Figure 2.4: The left hand side is the sketch of the actual size of H together with the three expected distances referred to the outer, inner and middle corner, respectively as well as the right hand side, which are the twelve centroids of the actual corners.

The algorithm extracts the helipad mark by analyzing the intrinsic properties of each connected components (i.e. blobs) inside the image such as the location of its center of mass or the area to perform cascade check. Later, the new algorithm tries to use the radius-of-curvature computation for each pixel belonging to the helipad contour in detecting the corner of helipad and tries to match the corner detected with those from the reference system. This method tries to overcome the limitations (i.e. computationally expensive). Indoor and outdoor experiments test the algorithm and it claims that there are small errors in the distance and angle estimations, which accounts for 1 % and 4 %, respectively.

The researchers use template matching and segmentation algorithms to detect and track a proposed remote marker design to facilitate UAV landing procedures for various time and condition (i.e. during morning, afternoon, evening and night) (Nguyen et al., 2017). The proposed remote marker design, having concentric circles features as shown in Figure 1.2, is to enable easy identification by vision algorithm at a high frame rate, irrespective of the distance between the UAV and the ground.

For daytime video, the algorithm includes image acquisition by the camera, image preprocessing by reducing the size of the image by a factor of four, template matching to recognize the marker and profile- checker to obtain the center position of the proposed remote marker design. It also uses Kalman filtering to predict the detected center and utilizes new measurements to correct or update these predictions.

For nighttime video, there is an incorporation of adaptive thresholding and morphological processing algorithm. Different from conventional thresholding that uses a global threshold for all pixels, adaptive thresholding changes the threshold dynamically over the image. In this literature, the threshold depends on the brightness histogram of the Region of Interest (ROI). The morphological processing uses the Hit-and-Miss algorithm to detect the pattern using a structuring element containing 1's, 0's and blanks for "don't cares". Figure 2.5 shows the result of the algorithm.

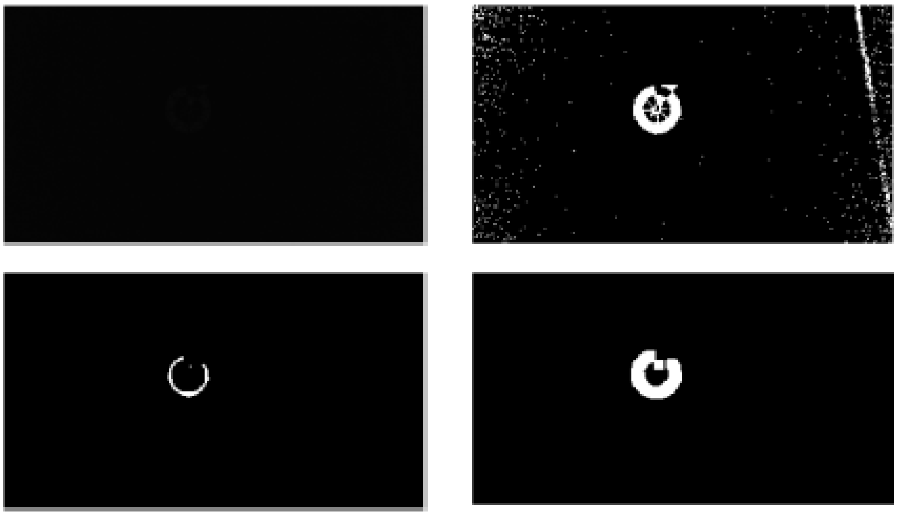


Figure 2.5: The top left is the original image at nighttime, top right is after adaptive thresholding, bottom left is after applying Hit and Miss morphology and the bottom right is the final result, respectively.

The researchers test the algorithm in various environments and claims that it outperforms the state-of-the-art visual trackers in terms of both robustness and accuracy. They recommend adopting a training scheme, specifically the deep learning-based tracking algorithm in the future.

Yet another example is an optical target based on concentric circles being the subject of detection to carry out the position and orientation determination algorithm optimized for high distances compared to the target size (Cucci, 2016). The proposed optical target, as shown in Figure 2.6, mainly makes use of concentric circles and white dots in the black ring encoding a signal. The reason to use circles instead of corner features is that circle projections, eclipses on the image plane can be accurately determined.

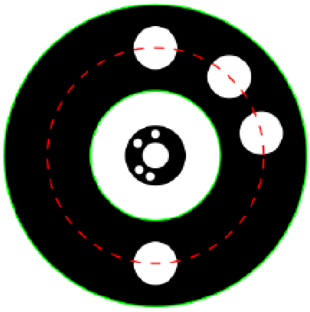


Figure 2.6: The optical target designed based on concentric circles and white dots in the black ring encoding a signal represented by the red dotted lines.

In summary, the algorithm includes mainly the target detection part: (1) contours detection, (2) circles detection, (3) code test and target heading determination and (4) sub-pixel edge detection and ellipse fitting as well as target pose estimation part. The (1) in the target detection part isolates edges (i.e. pixels with high intensity gradient), (2) uses a cascade check to identify circles out of all contours, (3) checks for the presence of the target code and (4) tries to locates the white dots in the black ring.

The analysis on aerial images taken by UAV elevated to a range of 10 m to 100 m proves that the algorithm works with accuracy. With a target size of 90 cm, the distance RMS is below 0.25 % under 50 m and below 1 % under 100 m. The author recommends obtaining further improvements by correcting for exposure-induced bias.

Another literature also describes the implementation from features detection to the fusion of visual and inertial measurements in performing vision-aided inertial navigation for small-unmanned aerial vehicles in GPS-denied environments (Wang et al., 2013). Generally, the algorithm includes detection of features, matching of features and fusion of visual measurement to that of IMU.

To detect the features, the researchers use a scale invariant features transform (SIFT). With this, there is an establishment of descriptor, which is invariant to rotation, translation and scale. Since the features in an image is enormous, there is a proposal of fuzzy-based threshold adjustment method to stabilize the number of features. This method reduces the calculation of the image processing without affecting the accuracy.

Then, there is a calculation on Euclidean distance between associated normalized descriptors. To match the features quickly, there is an application of a fast nearest-neighbour algorithm. If the distance between the two descriptors fulfills a set of criteria, then it is a matched pair.

The algorithm later fuses the visual measurement with the output of an IMU by an indirect EKF. In this literature, there is a proposal on a novel delay-based measurement update method, which can introduce visual measurement into the navigation state filter without any states augmentation.

There is a set-up of experiments in simulation to compare the proposed method to the general augmented methods and in real to verify its effectiveness. In the former, the results indicates that the proposed method can reduce the computational complexity by about 20 % and the latter have been verifying the effectiveness of the algorithm.

Another related work is the vision-based target finding and inspection of a ground target using a UAV system (Hinas et al., 2017). The target in this literature is a simple red circle. Presented is a vision-based algorithm, which focuses on target detection and navigation.

For target detection, the researchers use Hough circle or colour detection. However, after some trial and error, they find out the first method gives bad results with increasing altitudes while the second method gives unreliable results with decreasing altitudes. That is why, finally they use a combination of Hough circle and colour detection method to identify the red circle marker.

In the literature, the researchers make the UAV to hover above the target to perform close inspection. Figure 2.7 illustrates the process. To do that, they convert the 2D image coordinates to obtain the world coordinates that are useful in navigating the UAV by applying a few equations.

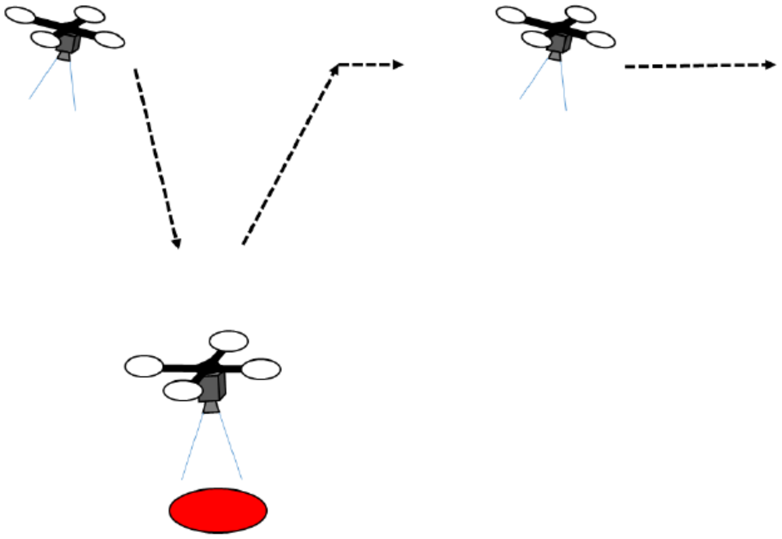


Figure 2.7: It shows the process of target detection and inspection.

Tested the algorithm in simulation and real experiment, the researchers claim to have a 100 % success rate for the simulation and 92 % for the real experiments. This is due to the challenges imposed by external disturbances. Two flight tests, which fail, are when the UAV is at an altitude of 15 m and 30 m, respectively. They recommend extending the system into one, which can detect multiple targets.

In addition, there is UAV visual autolocalization based on automatic landmark recognition (Filho et al., 2017). The researchers propose a new real-time CV position estimator for UAV navigation. The estimator uses images captured during flight to recognize specific or well-known landmarks, shown in Figure 1.3, in order to estimate the latitude and longitude of the aircraft.

The researchers test the algorithm in a simulated environment using a data set of real aerial images obtained in previous flights, with synchronized image, GPS and IMU data. From the qualitative evaluation, shown in Figure 2.8, it seems that the position obtained from the landmark recognition system is more accurate. The researchers recommend performing in-flight experiments in the future.

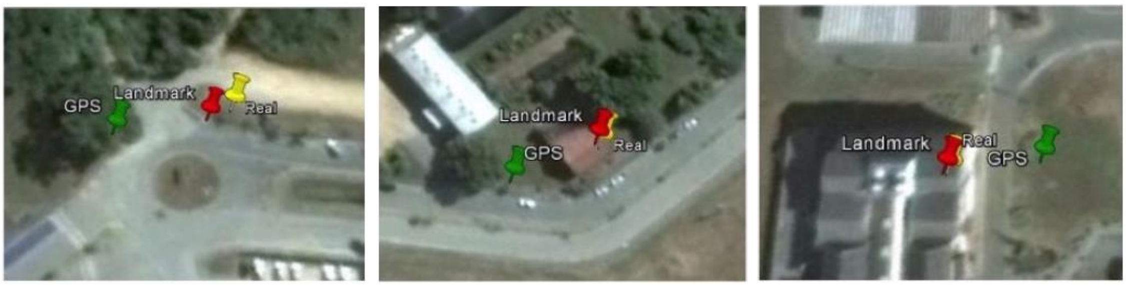


Figure 2.8: Auto-localization position comparison between GPS, landmark recognition estimation and real UAV position for the roundabout (left), roof of a house (middle) and roof (right), respectively.

There is also another example, which uses a vision-based guidance system for UAV navigation and safe landing using natural landmarks (Cesetti et al., 2010). In this literature, a features-based image-matching algorithm, alternately with GPS, locates the natural landmarks and gives feedbacks to onboard, hierarchical, behavioural-based control system. There is also a proposal of two algorithms for safe landing area detection based on features optical flow analysis.

To locate the natural landmarks, the algorithm starts with dividing the aerial image into different sub-images. Later, the authors apply SIFT, which gives distinctive, invariant to translation, rotation and scale, as well as robust against occlusion features, to each sub-images. If the features are useful, the algorithm proceeds with the features extraction.

The two algorithms for safe landing area detection are for (1) pure translational motion at a constant altitude and (2) pure descending motion at a fixed position. For the first case, the researchers use the SIFT features obtained previously to estimate the optical. Then, there is a reconstruction of structure, followed by a classifier to determine whether the landing area is safe or not. The second case requires the computation of a graph of distances between every features couple and compare the variation of this array of distances. Hypothetically, a geometric linear zooming over a flat surface brings to constant distance variations. That is to say, high variances bring to classification of unsafe area while very low variance gives us a feedback of the flatness of the landing area.

The researchers use a complete MATLAB/Simulink model to evaluate the feasibility of vision algorithm, which is based on features extractors and matching, by running it synchronously with the simulation of helicopter's dynamics. In addition, there are real experiments to test for the effectiveness of the safe landing area detection algorithms. Figure 2.9 shows the visual result of the simulation whereas Figure 2.10 shows that of the real experiments.

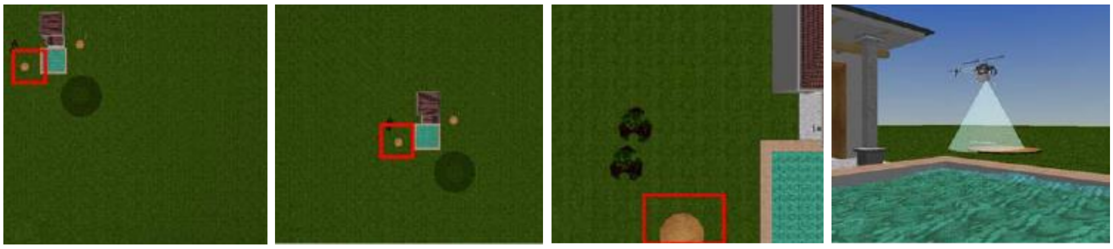


Figure 2.9: Starting from left to right, the leftmost is the initial frame when the vision algorithm detects the features, followed by the UAV reaches the target and it attempts to descend a little as well as the rightmost simulated 3D environment, respectively.

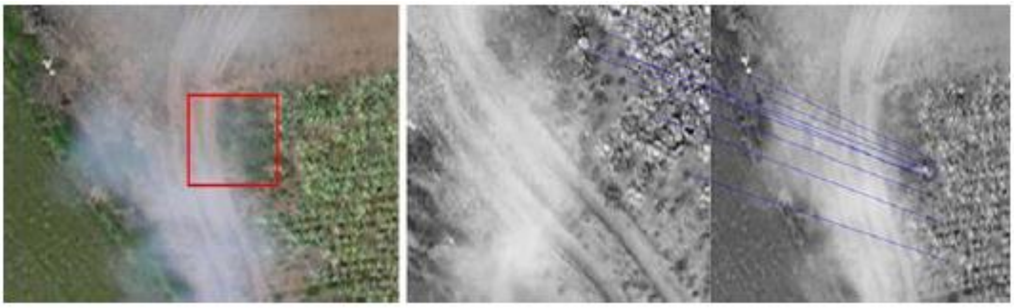


Figure 2.10: It shows the tracking of the landing area with respect to the template, using a natural landmark and dealing with smoke occlusion.

The researchers recommend improving the robustness of the whole system, specifically safety concerns during the landing task. They also hope to embark on achieving the capacity of the UAV to actively search for instead of just evaluating a safe landing zone.

Another research work that provides a UAV model in SIMULINK is (Prouty, 2005) (Pounds et al., 2010). It is a model based on the PARROT® series of mini drones (Prouty, 2005) (Pounds et al., 2010). It serves as a template for researches who wish to implement the flight simulation application. The UAV model is fully equipped with controllers for pitch, roll, yaw and height as well as sensors including an IMU, a camera and a sonar sensor as well as. The Simulink 3D Animation presents the model in an environment using 3D graphics objects (HUMUSOFT s.r.o, 2018). This project makes use of the platform to provide a simulation of UAV tracking the designed visual marker. Section 3.3 describes about the UAV flight simulation model in details.

CHAPTER 3

METHODOLOGY

3.1 Proposed Visual Marker Design and Its Features

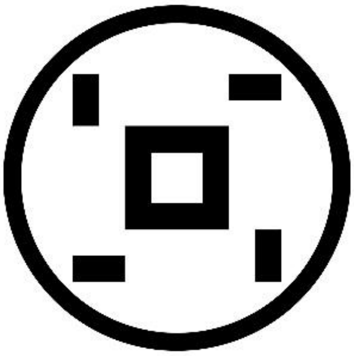


Figure 3.1: It shows the proposed visual marker design.

Figure 3.1 shows the proposed visual marker used in this work. Its main features include a black circle ring and a black square box, which are concentric as well as four rectangles oriented in between.

The inner circle has a diameter of 18 cm while the outer circle has a diameter of 20 cm. The dimension (i.e. width and height) of the inner square is 3 × 3 cm whereas the dimension of the outer square is 6 × 6 cm. In between, the rectangle, which locates on top left and bottom right, has a dimension of 1.5 × 3 cm while the rectangle, which lies on top right and bottom left has a dimension of 3 × 1.5 cm.

The idea to use a combination of circular and squared feature in proposing the visual marker design comes from reviewing literature, which uses an artificial visual marker. Normally, an artificial marker is either circular or squared. A circular visual marker is easily detectable at high altitudes and robust against occlusion while a squared visual marker has distinctive corner features, which are exploitable in the CV algorithm. The combination of both features in the proposed visual marker design is an attempt to make use of the advantages of respective features.

3.2 Designed Visual Marker Detection

The algorithm below aims to detect the designed visual marker in section 3.1 and provide its 2D pose with respect to the camera. The main steps are as followed: (1) image acquisition and preprocessing, (2) designed visual marker identification, (3) visualization of designed visual marker identification. Figure 3.2 shows the summary of the algorithm.

```
Algorithm 1: Designed Visual Marker Detection
Input: Image
Output: Location of artificial visual marker in the image
    1. Preprocess the image by turning it into grayscale and later binary
    2. Determine the properties of image regions by using regionprops function
    3. Calculate the distance between centroids
       If distance between two centroids is in a range of interest
    4. Record the centroids fulfilling the condition
       end
    5. Obtain the ratio of diameters of adjacent image regions
       If ratio of diameters of adjacent image regions is in a range of interest
    6. Record the diameters fulfilling the condition
       end
    7. Draw circles to visualize the image regions identified to be the marker
```

Figure 3.2: It shows the algorithm in detecting artificial visual marker.

3.2.1 Image Acquisition and Preprocessing

The monocular camera does the image acquisition. After acquiring the image, it is typical and essential to preprocess it to ease the analysis in the later stage. Preprocessing an image enables the removal of noise effects (i.e. unwanted objects) and highlight image details (e.g. object edges).

When first acquired, the image is in its truecolour, more popularly known as Red-Green-Blue (RGB); it is an m-by-n-by-3 data array, which defines each colour components. RGB refers to a system to represent colours. Any numbers ranging from 0 (i.e. off) to 255 (i.e. brightest) define each colour components. The combination of each component in various proportions give rise to any colours in the visible spectrum. That is to say, the total number of available colours is 256 × 256 × 256 or 16777216 colours! Table 3:1 shows some of the colours with the components stated. To get a pure red colour, set the red number to the maximum and zero for both the green and blue number. To get a pure white colour, set the three components number to the maximum. Do the opposite to get a pure black colour.

Table 3:1: It shows the colours and the components stated.

Colour	Red Number	Green Number	Blue Number
Red	255	0	0
Green	0	255	0
Blue	0	0	255
White	255	255	255
Black	0	0	0

In MATLAB, applying the *imread* function helps to read an image from graphics file in RGB. Given the complexity of a RGB image mentioned above, there is a need to preprocess it before any further analyses. One of the methods is to convert RGB into grayscale.

Grayscale is a range of monochromatic shades from black to white, giving luminance (i.e. intensity) value. In other words, it contains only shades of gray and no colours. The intensity value ranges from 0 (i.e. black) to 100 (i.e. white). Converting RGB to grayscale means the removal of colour information, retaining only the luminance information. That also means the m-by-n-by-3 data array, now, changes to an m-by-n-by-1 data array, which is easier to process. Equation 3.1 converts RGB to grayscale.

$$Luma = 0.299R + 0.587G + 0.114B \tag{3.1}$$

However, inside MATLAB, there is already a built-in function to do the conversion. It is the *rgb2gray* function.

To take a further step in preprocessing the image, convert the grayscale into binary. Each pixel in a binary image assumes one of the two discrete values: 1 or 0; the former represents black colour while the latter white colour. Binary image forms by thresholding a grayscale image. Binary image is the simplest to process. In MATLAB, the function to binarize a grayscale image is *imbinarize*. After preprocessing the image, the algorithm now proceeds to identifying the designed visual marker, explained in details in the following subsection.