# FAST 3D TRACKING ALGORITHM IMPLEMENTATIONS ON MINI QUAD-COPTERS

A Thesis

by

Şeyma Sivritaş

Submitted to the

Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Electrical and Electronics Engineering

Özyeğin University
July 2020

# FAST 3D TRACKING ALGORITHM IMPLEMENTATIONS ON MINI QUAD-COPTERS

Approved by:

Asst. Prof. Ahmet Tekin, Advisor,
Department of Electrical and
Electronics Engineering
*Özyeğin University*

Asst. Prof. Göktürk Poyrazoğlu,
Department of Electrical and
Electronics Engineering
*Özyeğin University*

Asst. Prof. Reyat Yılmaz,
Department of Electrical and
Electronics Engineering
*Dokuz Eylül University*

Date Approved: 20 July 2020

*To my family…*

# ABSTRACT

In recent years, unmanned aerial vehicles (UAVs) have been integrated into many applications. These vehicles are used by the governments and the private sector for many different purposes. Some of the popular applications are field scanning, target detection, target tracking, surveillance, defense markets, search and rescue, map creation, mine, water and underground resource search, photography or entertainment. Recently many academic institutions have shown significant amount of interest in this field and many of them have established aviation departments and societies. In addition, many thesis studies and academic publications related to UAVs have been presented. The aim of this particular study is to detect a small target object by UAV and to provide autonomous tracking of the detected target in three axes. As the outcome of this work, the UAV is coded to lock on the target and maintain a fixed distance on the x, y and z axes after the initial lock. Within the scope of the study, a functional system was designed by optimizing important parameters for UAVs. An algorithm was designed that uses the image processing methods and utilizes MAVLink communication protocol. Image processing was used to detect the target, detect the movement direction of the target and calculate the movement distance of the target. MAVLink communication protocol was used to control the behavior of the UAV in x, y and z axes. In addition, two distinct methods such as the algorithm method and using an external distance sensor were tested to keep the distance constant on the z-axis (vertical) and reported. The project was successfully implemented using single camera and an ultrasonic sensor.

**Keywords:** Unmanned aerial vehicle, UAV, autonomous systems, 3D tracking, image processing, fast algorithm.

# ÖZET

Son yıllarda insansız hava araçlarının (İHA) kullanımı birçok uygulama alanına entegre edilmiştir. Bu araçlar devletler ve özel sektör tarafından farklı amaçlarla kullanılmaktadır. Popüler olan uygulamalar arasında alan tarama, hedef tespiti, hedef takibi, gözlem, savunma sanayi, arama kurtarma, haritacılık, madencilik, su ve yer altı kaynakları araştırması, fotoğrafçılık ve eğlence alanları sayılabilir. Son zamanlarda birçok akademik kurum bu alana büyük ilgi gösterdi ve birçoğu tarafından havacılık bölümleri ve toplulukları kuruldu. Ayrıca insansız hava araçları ile alakalı birçok tez çalışması ve akademik yayın hazırlanmıştır. Bu çalışmanın amacı ise küçük bir hedef nesneyi insansız hava aracı ile tespit etmek ve tespit edilen nesnenin otonom olarak her üç eksende de takip edilmesini sağlamaktır. Gerçekleştirilen çalışma ile insansız hava aracı, hedefi ilk kez tespit ettikten sonra x, y ve z eksenlerindeki mevcut mesafesini koruyarak hedefe kilitlenecek şekilde kodlanmıştır. Çalışma kapsamında insansız hava aracı için önemli parametreler değerlendirilerek fonksiyonel bir sistem tasarlanmıştır. Görüntü işleme metodunu ve MAVLink iletişim protokolünü kullanan bir algoritma da tasarlanmıştır. Hedefi tespit etmek, hedefin hareket yönünü tespit etmek ve hedefin hareket mesafesini hesaplayabilmek için görüntü işleme metodu; insansız hava aracının x, y ve z eksenlerindeki davranışını kontrol edebilmek için ise MAVLink protokolü kullanıldı. Ayrıca, z eksenindeki (dikey) mesafeyi sabit tutmak için algoritma yöntemi ve harici mesafe sensörü kullanımı gibi iki farklı yöntem test edilmiş ve raporlanmıştır. Proje, tek kamera ve ultrasonik sensör kullanılarak başarıyla gerçekleştirilmiştir.

**Anahtar Kelimeler:** İnsansız hava aracı, İHA, otonom sistemler, üç boyutlu takip, görüntü işleme, hızlı algoritma.

# ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor Asst. Prof. Ahmet Tekin for his patience, understanding and advice.

I am also grateful to my family for their trust, patience and support during this period.

Şeyma SİVRİTAŞ

# TABLE OF CONTENTS

# LIST OF FIGURES

# CHAPTER I

# INTRODUCTION

We can define the unmanned aerial vehicle as an aircraft that does not have a pilot and can fly on its own with its automation and communication technologies. As a result of the increase in R&D investments for this field in last decades, unmanned aerial vehicles (UAVs) have become quite widespread and popular in civil and commercial fields, especially in military fields. After the 1990s, developments in the computer processors, sensors and communications industry accelerated the developments of unmanned aircraft technology. These vehicles may be deployed for many different purposes. The main use cases for these technologies are intelligence, surveillance, assault, search and rescue, early warning, air defense, cargo transportation, map creation, agriculture, photography, advertisement or entertainment. UAVs are also popular topic for academic research [1][2][3][4].

The UAVs have important advantages over the manned vehicles. Some of these advantages are:

- Low maintenance and operating costs,
- Longer flight time,
- Unlimited flight time due to flight crew,
- Possibility of reassignment in the air,
- Minimum risk of human error,

- Flight opportunity to work without risk of loss of team in dangerous and dirty environments,
- No risk flight crew loss in case of accident

The UAVs have some advantages over satellites as well. These are:

- Low supply, maintenance and operating costs,
- The data obtained is more accurate and of higher quality (like image resolution),
- Continuous coverage of a target area,
- Higher maneuverability [5].

The UAVs are classified according to some characteristic features such as weight, altitude, endurance, range, etc. [6].

The UAV used in this thesis study is in "Mini UAV" category with respect to mini UAV limits shown below [7];

- Maximum takeoff weight: < 30 kg.
- Maximum flight altitude: 300 m.
- Endurance: < 2 hours
- Data link range: < 10 km.

## *1.1 History of the UAVs*

The first unmanned aircraft was designed by Elmer Sperry and Peter Hewitt at the end of the First World War (1916) for the US Armed Forces. The first unmanned use was carried out with the "Hewitt-Sperry" gyroscope controlled automatic aircraft known as "aerial bombs". This aircraft was stabilized with 12 horsepower motor when the weight is 175 kilograms. It was capable of measuring under 19 feet for determining altitude thanks to gyroscopes and barometer [8].

In the 1920s, Etienne Oehmichen experimented with rotorcraft vehicles. One of the quadcopter is named as "No.2" which has four engines and eight propellers as shown in Figure 1. In the middle of 1920s, Mr. Oehmichen carried out more than a thousand test flights. No.2 remained 7 minutes and 40 seconds in the air. Moreover, it flew 360 meters and broke the distance record [9].

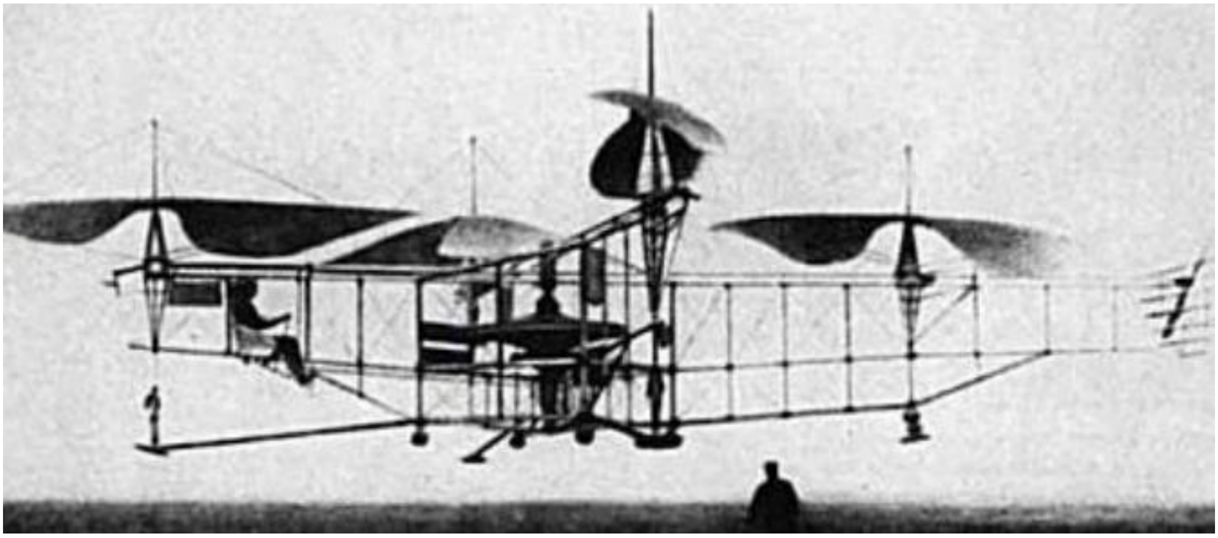

**Figure 1: Oehmichen No.2 [9]**

The first radio-controlled UAV was designed by Reginald Denny in 1933. Many various drones were produced during World War II. The Army improved this aircraft named as OQ-19 and produced about 48,000 pieces for serving during World War II. It was used as the world’s most popular target drone [10].

At the end of the 1930s, the Curtiss N2C-2 was developed by US Navy as a target drone. This drone was remotely controlled by another aircraft. The Curtiss N2C-2 was used to attack a target ship by US Army Air Force [11]. Curtiss N2C-2 can be seen from Figure 2.

**Figure 2: Curtiss N2C-2 [11]**

In 1942, TDN-1 torpedo assault remote-controlled drone was improved by the US Navy to carry a torpedo or a bomb of mass about 900 kg while the airspeed of drone is 280 kmph. Although this drone was lightweight, cheaper and has simpler design, total production pieces are not much. This TDN-1 drone was not used for real battle [11]. TDN-1 drone is shown in Figure 3.

**Figure 3: TDN-1 Torpedo Assault Drone [11]**

In 1944, TDR-1 Assault drone was designed as upgraded version of TDN-1 assault drone. It was improved by using new technology such as mounting sensor that is RCA television camera of mass about 35 kg on front side of the drone. The 140 pieces of drone were produced and it was used against the Japanese [11].

In 1960, AQM-34 Ryan Firebee was developed by U.S Air Force. It was first stealth aircraft and used for reconnaissance missions. This drone achieved great success over Southeast Asia and Vietnam War [12].

In 1970s, “Scout” was produced by Israel Aircraft Industries (IAI). This is the first operational Unmanned Aerial System of IAI. It was used for surveillance and target acquisition in 1982 Lebanon War. The main missions of the Scout were intelligence, surveillance and reconnaissance (ISR). It was used to detect and listen to the enemy's communication and allow communication of friendly troops with each other [13].

The Tu-141 was developed by Tupulev Design in the middle of 1970s. Tu-141 and its variants were commonly used by Russia. This drone had one-thousand kilometers range and it had ability of released from a trailer. [14].

In 1992, design of a very small micro-drone (now micro air vehicle) was planned by Defense Advanced Research Projects Agency (DARPA). As a result of these studies, it is determined that the realization of many micro technologies on these micro vehicles is possible. Maximum 15 cm. of size and maximum 90 grams of weight parameters were set as target. Draft design was drawn by R.J.Foch in 1995 as shown in Figure 4. Main goals of this micro-drone were providing detection, tracking and communication [15].

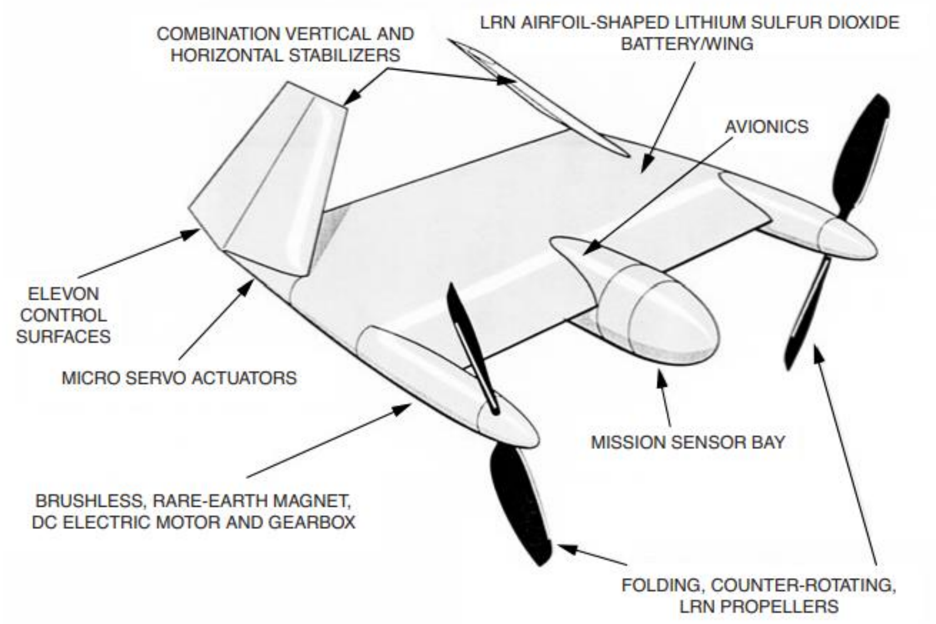


**Figure 4: Micro air vehicle baseline design [15]**

In 2000s, X-45 unmanned combat air vehicle (UCAV) was developed by Boeing Company and it is commonly used by US Military [16]. The combat air vehicle has capability of air-to-air and air-to-ground attack [17]. This UAV is shown in Figure 5.


**Figure 5: Boeing X-45 UCAV [16]**

## *1.2 Aim of the Study*

The aims of this thesis are to design a 3D point tracking algorithm and implement this designed algorithm on a quad-copter in order to obtain an autonomous unmanned air vehicle system. Within the scope of this thesis, it is intended to detect the point source target and start 3D tracking of this target until the operator finished the mission.

# CHAPTER II

# PREVIOUS WORKS

Since the UAVs and autonomy are very popular in recent years, many projects were implemented in these areas. Some of the studies were examined and summarized below.

In 2013, ground object tracking problem was handled by using a ready consumer drone only on x and y axes. There were 2 cameras located on front and bottom sides of the drone and they have 320x240 and 176x144 pixels resolution, respectively. The flight duration of this drone was 13 minutes and the speed was more than 5 m/s. PD (Proportional-Derivative) algorithm was used for control of drone in this project. The tracking performance was controlled by pixel size comparison and color sensing method was used for object detection on the captured images on the host computer. It was observed that this method is very sensitive while working in outdoor conditions due to noises such as insufficient camera quality, daylight, etc. It is planned that more efficient target detection methods will be handled in the future works [18].

Another study was performed by a group about visual based object tracking without GPS using visual tracking and target model learning algorithm in 2013. The algorithm was run on a consumer drone in outdoor conditions. The stable or moving objects were tracked thanks to OpenTLD (Tracking Learning and Detection) algorithm. All processes were done in an external computer and control of the drone was provided by this computer via wireless communication. [19]. In this project, the single camera was used for object tracking. Since the system has no ability of determining depth, the distance between target and drone was kept constant [19].

In 2014, autonomous 3D tracking project was performed. The external computer was used to reduce power consumption. Two cameras were integrated into drone horizontally and vertically for this project. Altitude was measured by ultrasonic sensor. The camera resolution was 176 × 144 pixels. Control mechanism of quad-copter was provided by PID controller and this project was realized in indoor environment [20].

In 2015-A project was previously implemented for tracking an object by a drone. In this study, two embedded cameras were used for object detection and tracking. A TLD (Tracking-Learning-Detection) algorithm was used for tracking ability [1].

There is another prominent work in the literature about autonomous flight in 2015. A reinforcement learning method was also applied in this work. In addition, a camera system, which consists of 24 cameras, was used for this study [21]

In 2016, a group worked on real time object tracking project and the project was successfully performed. In the project, a special motion capture camera system was used to detect the position of the moving object and Matlab simulink was used to process the information. In addition, a computer was used as a base station [22].

The other project is about detection of target object and 3D tracking of the target autonomously by using a consumer drone in 2016. The project is implemented in indoor conditions. For tracking of 3D moving targets, the clustering of Static-Adaptive Correspondences for Deformable Object Tracking (CMT) algorithm was used because of more effective algorithm according to OpenTLD (Tracking Learning and Detection). The operation of CMT tracker was observed by using IBVS (Image Based Visual Servoing) controller method. Even if the targets are deformable, the algorithm still has tracking ability thanks to these algorithms. The future work was planned to achieve same mission in outdoor conditions to show effectiveness of this project [23].

In 2019-In another similar project, visual detection and tracking implementation was realized with UAVs. In this project, Haar cascade classifier was used to detect the object and two cameras were used to achieve the target function [24].

Within the scope of this work, a programmable autonomous UAV system was designed, and a mission was proven successfully. The mission was detection of a target object and 3D (3-dimensional) tracking (on x, y, z axes) of the target autonomously. The study was implemented by using a GPS and a single camera that has 640 x 480 pixels resolution, a distance sensor for obtaining depth information and hence a simpler algorithm by taking into account factors such as cost, weight and ease of application. The flight duration of the drone is more than 30 minutes. PID algorithm was used for control mechanism of the drone and this project was realized in outdoor environment conditions. Allowing the implemented system to be upgraded with simple modifications is an important advantage of this project for developing the project in the future.

In this paper, main components of the autonomous UAV system are listed and described. Also, design steps of the image processing subsystem were summarized. The method of detecting the position of the moving target and calculation of the updated coordinates according to displacement of the target were presented. Two different techniques for altitude tracking were analyzed and the results were compared.

# CHAPTER III

# SYSTEM PLATFORMS

Overall system consists of two main platforms such as hardware and software platforms.

## *3.1 Hardware Platform of the System*

A UAV was used with SDK (Software Development Kit) flight controller to achieve the mission. Some auxiliary components (sensors) were also used to complete the autonomous system design. These components are:

- **GPS** was used to determine the located coordinates of the UAV.
- **Compass** was used to determine the directions.
- **Accelerometer** was used to determine the acceleration in three axes (x, y, z).
- **Gyroscope** was used to determine the motion in three axes (x, y, z).
- **Barometer** was used to determine the altitude by using pressure value.
- **On-board computer** was used to provide the communication with the flight controller of the UAV for behavior commands. It was also used as hardware platform for image processing software.
- **Camera** was used to get the image for image processing.

- **Distance sensor** was used to provide altitude tracking.

### 3.1.1 Flight Controller

Flight controller is the circuit that allows the drone to fly stably and provides sensing capabilities to UAV. Pixhawk flight controller was preferred for this project. Pixhawk flight controller is one of the most popular low cost and high performance autopilot solution [25][26].

Pixhawk is an open source and open hardware toolkit that has advanced 32-bit STM32F427 ARM Cortex dual microprocessor, 14 PWM servo outputs, $I^2C$ and CAN bus serial communication interfaces. Moreover, Pixhawk includes dual compass, GPS, accelerometer and gyroscope, etc. [25]. These sensors are used dual in order to provide redundancy management and accuracy of measurements. The flight controller also allows flight data records to be stored in the SD card [27]. Pixhawk flight controller can be seen from Figure 6.


**Figure 6: Pixhawk flight controller**

The used STM32F427xx microprocessors are part of the STM32F4 family and these devices are based on the high-performance ARM® Cortex®-M4 32-bit RISC core with FPU (Floating Point Unit). Operating frequency can be increased up to 180 MHz [28]. The current consumption of the processor is 244µA/MHz in run mode and it corresponds to 44 mA at 180 MHz. The consumed current is 2.2 µA - 3.1 µA in standby mode and it is smaller than 1 µA in $V_{bat}$ mode [29].

### 3.1.2 System Computer

A computer was used for image processing and communication with flight controller. It is also used to download the telemetry data to base station. Raspberry Pi 3 Model-B was preferred for this study because it can easily perform most of the features fulfilled by desktop or laptop. Raspberry Pi 3 Model-B has Quad-Core ARM Cortex-A53 (64Bit) and Broadcom BCM2837 chipset. The processor speed can be increased up to 1.2 GHz. Raspberry Pi has CSI camera port which allows easy connection with Raspberry pi camera. Even though it is a credit card sized computer, it offers the opportunity to work with high resolutions as 1920 x 1080 pixels resolution. This small dimensions and lightweight features are crucial advantages to provide integration the system computer into UAV. Because, larger equipment causes the system increase in size and heavier part increases the weight of the system, hence cause more power consumption. In addition to these features, Raspberry Pi can also connect to wireless devices with Wi-Fi or Bluetooth connectivity. In this way, it is possible to connect to the wireless network inside the network region and observe the computer operation from base station [30]. This feature was often used during the development stage. Raspberry Pi 3 Model B is shown in Figure 7.

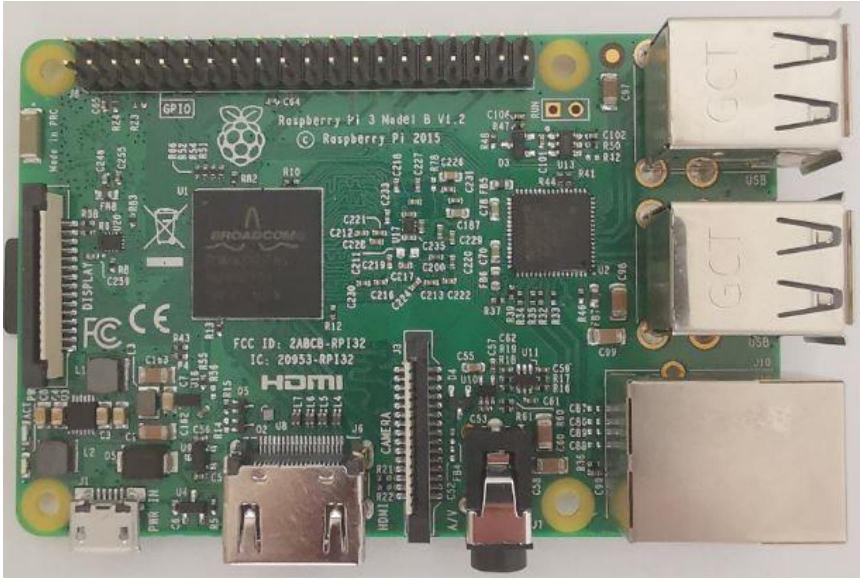

**Figure 7: System computer (Raspberry Pi 3 Model B)**

### 3.1.3 Camera Module

Since the Raspberry Pi Camera Module is naturally supported by Raspberry Pi 3 Model-B, this camera was chosen for target detection and tracking missions. This camera is lightweight (3 grams) and compact. The dimensions of the camera are 25 x 24 x 9 mm as shown in Figure 8. It supports FHD (1080p) video stream. Horizontal and vertical angle of view of the camera are 53.50°±0.13° and 41.41°±0.11°, respectively. These angles play an important role in image capturing process during the mission. This effect will be explained in detail in the next sections [31].

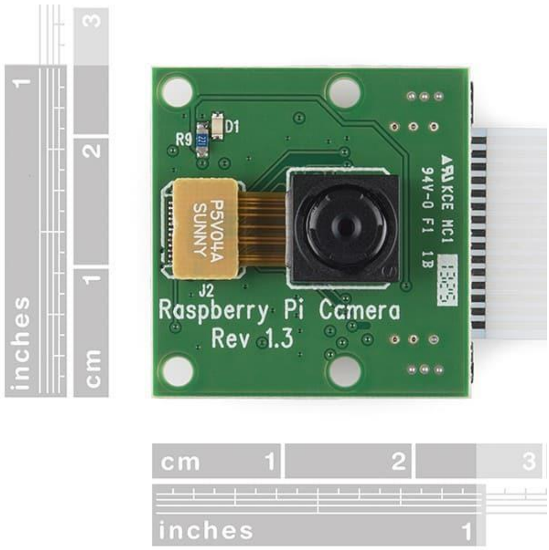


**Figure 8: Raspberry Pi camera module [32]**

### 3.1.4 GPS (Global Positioning System) Module

Satellites constantly send their location in the orbit and sensitive clock information to Earth. The GPS receiver must receive signals from at least three satellites for latitude and longitude information. If it is desired to have altitude information, $4^{th}$ satellites connection must be provided.

U-blox NEO-6M GPS module was preferred for this study and this GPS module can be seen from Figure 9. Having high accuracy performance, being cost-effective and lightweight are the other parameters to choose this module. Despite 2.7V - 3.6V input voltage range; the module is 5V tolerant and therefore no need to use external 3.3V voltage converter. This GPS module is very popular in UAV projects thanks to all these features [33].


**Figure 9: GPS module**

### 3.1.5 Telemetry System

Telemetry is a remote or wired monitoring or control of a part of any system that cannot be reached. Telemetry applications are realized with wireless communications generally. The application area doesn't need to be flat if wireless telemetry is used. Data signals can be transmitted even if obstacles such as buildings, hills exist in the field [34].

The various types of data which are provided by the sensors through the flight controller of an unmanned aerial vehicle were listed below. This data can be obtained and monitored from ground control station.

Information about the state of the UAV;

- The angles of Roll, Pitch and Yaw
- Altitude
- Horizontal and Vertical Speed
- Battery Status
- Battery, Motor, ESC and Electronic Circuit Temperatures

- Current Consumed by motors
- Motor RPM
- Signal Power
- Fault Warnings

Information about the environment where the UAV locates;

- GPS Coordinates
- Outside Temperature
- Pressure
- Wind
- Status of the task to be performed by the UAV [35].

During the project development process, 433 MHz wireless telemetry kit was used which includes;

- 2 pieces 433 MHz telemetry transceiver,
- 2 pieces 433 MHz antenna [36].

It is used for a telemetry connection between used drone and a USB connection supported device such as laptop, tablet, etc. in this project. Serial communication feature of the telemetry kit was used for data transfer between UAV.

The system utilizes the 433 MHz band and has 2 km range in case of proper weather conditions. The range decreases to around 300-500 meters in bad weather conditions. The telemetry set has 0-20 dBm (0-100 mW) adjustable output power and -117 dBm sensitivity. The used telemetry kit has small size and it is lightweight as preferred for UAV projects [37][38]. These telemetry modules are shown in Figure 10.


**Figure 10: Telemetry modules**

### 3.1.6 Voltage Converter (Battery Eliminator Circuit)

BEC (Battery Eliminator Circuit) module regulates voltage coming from LiPo battery and provides 5V output voltage. This voltage is used by flight controller, system computer and the other auxiliary components such as motors, sensors, etc. The used BEC module allows input voltage from 2S to 8S which equals to 7V and 30V, respectively. Here, “S” letter represents series connected cells of the battery and each cell equals 3.7V. The converter allows max. 90A current consumption. This module is quite small which has 2,5 cm x 2,1 cm x 0,9 cm and weighing 17 grams. The voltage converter board can be seen from Figure 11.

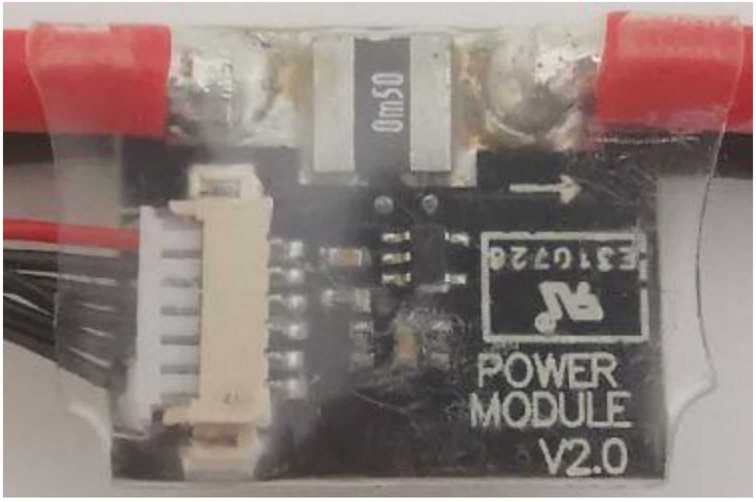


**Figure 11: Voltage converter board**

## 3.2 *Software Platform of the System*

### 3.2.1 Digital Image Processing

Digital image processing is rapidly growing with the increasing of technological studies [39]. It was used for many application areas such as bio-medical, space, aviation, robotics, etc. [40].

A digital image is a two-dimensional image or picture represented digitally with the combinations of bits 0 and 1 or specifically called pixels. Each pixel is encoded with 8 bits and each pixel consists of $2^8=256$ different gray scale equivalent.

If the image processing is applied for grayscale images, each pixel consists of a numerical value that is between [0, 255] range. Gray level 0 corresponds to pure black color and gray level 255 corresponds to pure white color. Other gray levels are placed between these two values.

If the image processing is applied for color images, three gray level values are required that are R (Red), G (Green), B (Blue) colors. Since each pixel is encoded with

8 bits, three colors (Red, Green, Blue) will be encoded with 24 bits and each pixel consist of $2^{24}$ (approximately 17 millions) different color values. The value of the combination of the three colors is shown such as RGB= (0, 0, 0), ..., (255,255,255) [41].

Digital image processing application was used for detection and tracking of the target in this thesis. Within the scope of image processing, Python programming language and OpenCV library were preferred.

#### *3.2.1.1 Python Programming Language*

Python is an interpreted programming language and doesn't need compiling process [42]. It is open source, object-oriented and the usage of this language is free for academic and commercial purposes. It is very simple against other programs and implementation of code is easy, so it can be learned quickly without wasting time [43]. Furthermore, the necessary code length is minimized for implementation with this language [44].

It can work on almost any platform like that Windows, Linux, Mac OS, mobile phones, embedded systems, etc. [43]. Python language was used for image processing with a library includes specific operations to facilitate the design in this project.

#### *3.2.1.2 OpenCV Library*

OpenCV (Open Source Computer Vision) is a library for programming languages and it was initially coded with C programming language [45]. It is open source and very helpful. This library provides significant support in coding and hence saves time [46]. There are more than 500 optimized algorithms for image processing and machine learning within the OpenCV library. In terms of usage, it has support for Linux, Windows, Android, MacOS [47].

OpenCV provides to target determination quickly and accurately for image processing operation [48]. Both of Python and OpenCV are widely used in an image processing area which involves logical & arithmetic operations [49].

#### *3.2.1.3 Color Space*

The image is captured in RGB color space from camera, but it is converted to HSV (Hue, Saturation, Value) color space to have more details about color such as saturation and light intensity. The most accurate HSV values can be obtained by experimentally masking the target in the mission environment [50][51].

#### *3.2.1.4 Gaussian filtering (Blurring)*

Gaussian filtering is a kind of low-pass filter and this function is used to blur an image. Sharp transition (high frequency) causes noise in the image. This filter prevents this sharp transition by blurring related part and provides smoothing. Hence, high frequency noises can be purged from the image with the help of this filter [46][52].

#### *3.2.1.5 Masking*

The desired object is masked from whole image by making the color of the target object white and color of the rest of the image black. This process is necessary to implement Morphological Transformations to be described in next section to obtain better quality from captured image by eliminating noises [50][51]. An example of masking operation is shown in Figure 12.

**Figure 12: Masking operation**

*3.2.1.6 Erosion*

One of the two fundamental mathematical morphological operations is erosion and it is typically applied to binary image that work on grayscale images. The Erode function is very useful for removing noises. This operation is applied for eroding boundaries of the foreground objects and as a result of this process, the foreground area is reduced from its outer edge to inside its area [53][54][55][56]. Since the pixels within the specified area according to the parameters are eroded, noisy input image is cleaned by removing it from the small noises. A simple application of erosion operation is shown in Figure 13.

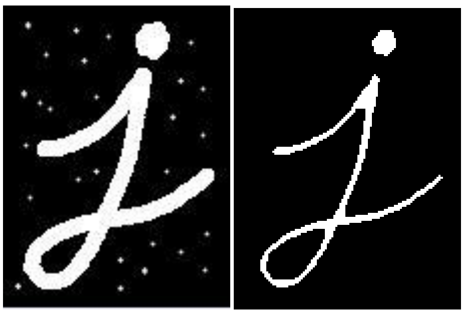

**Figure 13: Erosion operation [56]**

The erosion operator consists of two data inputs. One of them is the image that will be eroded, the other one is structuring element that determines the exact effect of the erosion on the input image [57].

The structure element can be defined as desired. The sample structure element is preferred as 3x3 matrix of ones and it can be seen in Eq. 1. Pixel version of the structure element is shown in Figure 14. Yellow pixel in this figure represents the center pixel of the structure element.

$$F_s = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1}$$

where,

$F_s$ = Structure element matrix

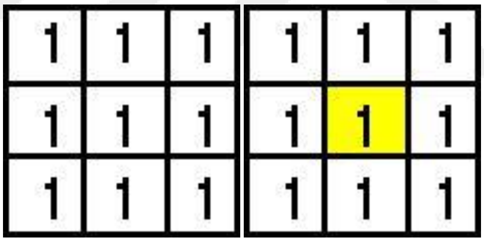


**Figure 14: Pixel views of structure element for erosion operation**

The structure element and related pixels of the input image are superimposed. The pixel to be processed is the pixel under the center of the structure element. If all of the pixels under structure element are “1” (foreground pixels) then the related pixel is left as “1”. If any of the pixels under structuring element is “0” (background pixel) then the related pixel is set as “0” [58]. Erosion process can be observed from Figure 15.

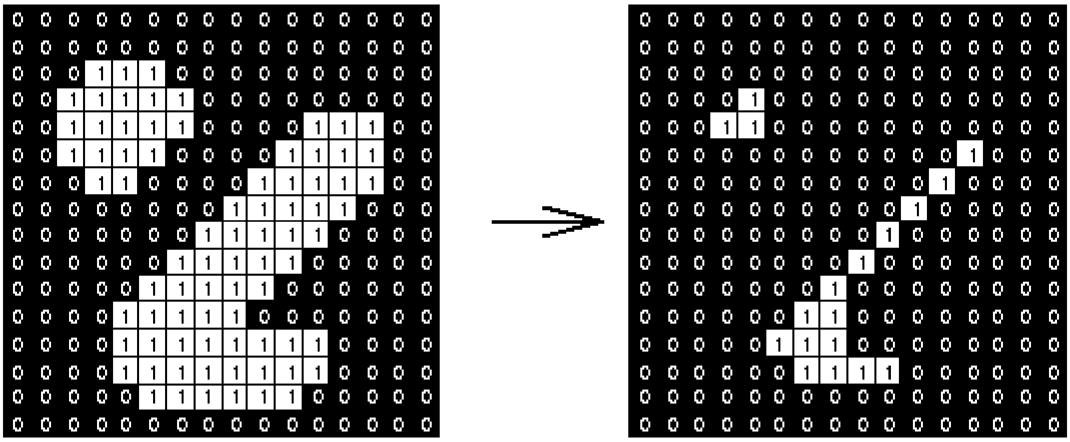

**Figure 15: Application of erosion process [59]**

To protect the size of the original objects, the "Dilation" function is used after erosion process [53][54][55][56].

### *3.2.1.7 Dilation*

The other mathematical morphological operation is dilation. Dilation operation is reverse of the erosion process. The dilate function is used to expand the boundaries of the objects and this function provides enlarging the area of the foreground objects. The Dilate function is generally used with "Erode" function, after erosion [53][54][55][56]. A simple application of dilation operation is shown in Figure 16.

**Figure 16: Dilation operation [56]**

The dilation operation consists of convolving an input image with some structure element [60]. The structure element can be defined as desired. Same structure element was used with Erosion process and it was preferred as 3x3 matrices of ones again. It can be seen in Eq. 2 and pixel versions of the structure element are shown in Figure 17.

$$F_s = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad (2)$$

where,

$F_s$ = Structure element matrix

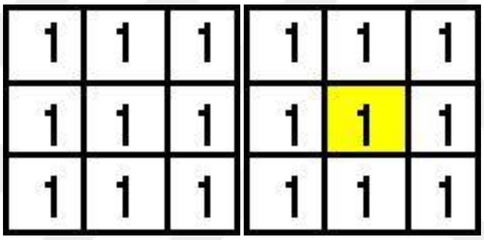

**Figure 17: Pixel views of structure element for dilation operation**

The pixel to be processed is the pixel under the center of the structure element. When the structure element is scanned over the image, if any of the pixels under structure element is “1” (foreground pixels) then the related pixel is set as “1”. If all of the pixels under structuring element are “0” (background pixel) then the related pixel is left as “0”. After this process, white region increases and size of the foreground object becomes larger [58]. Dilation process can be observed from Figure 18.

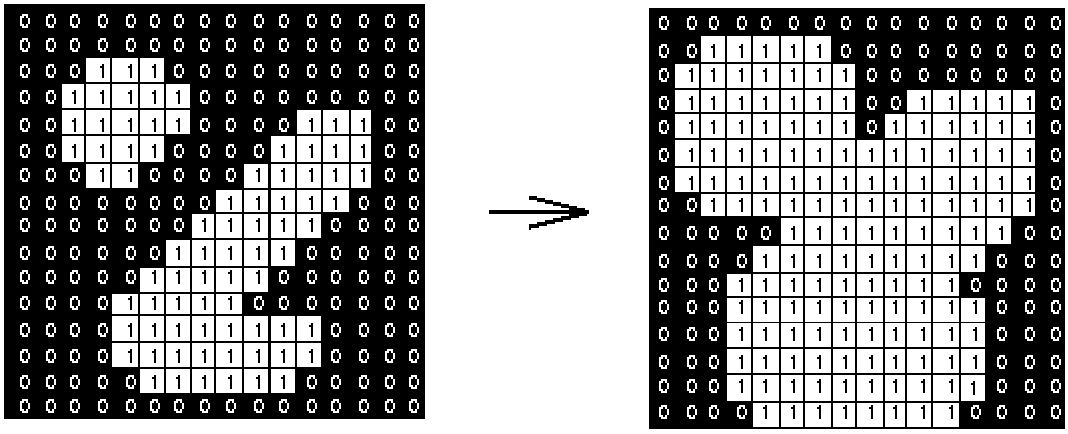

**Figure 18: Application of dilation process [59]**

After eroding and dilating operations image can be cleared from small noises as seen from Figure 19.

**Figure 19: Result of erosion and dilation processes for removing noises [56]**

### 3.2.2 Control Mechanism

PID (Proportional, Integral, Derivative) controller includes feedback control loop mechanism and it is used to improve the system characteristics and dynamic performance of the UAV [61]. Due to some physical, electrical or magnetic factors, the operation of the system may be negatively affected. The stabilization of the UAV was provided with the help of PID control mechanism integrated into flight controller, therefore it is very important to determine the most suitable gain parameters during control process [62].

The main task of PID algorithm is to minimize the error of the system dynamically, hence to minimize the difference between row, pitch, yaw angles that is received from sensors and desired angles that is taken by the user. The difference between a desired set point and a measured process variable is calculated by PID controller. As a result of this process, PID algorithm calculates the required motor speed (rpm) values according to this improving process. There are three parameters in PID controller calculation.

The block diagram of the PID controller is shown in Figure 20 and the calculation method is shown in Eq. 3.

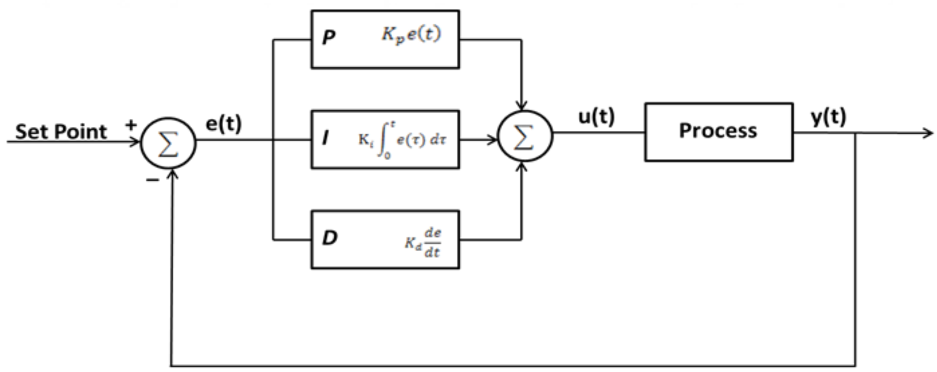


**Figure 20: Block diagram of PID controller**

$$u(t) = k_p e(t) + k_i \int_0^t e(\tau)\, d\tau + k_d \frac{de}{dt} \qquad (3)$$

where,

$K_p$: Proportional gain

$K_i$: Integral gain

$K_d$: Derivative gain

u: Control signal

e: Error

The first component is proportional control; it controls the actual value of the error. Proportional control proceeds based on the current angle error of the UAV [63][64]. $K_p$ coefficient shows the level of the correction to be applied to drone.

If a high $K_p$ coefficient is chosen, then the aircraft will try to further adjust its movement and the response would involve an "overshoot". The UAV will be very sensitive and will start to vibrate at high frequency.

If a low $K_p$ coefficient is chosen, then maneuvers of the drone will be corrected very quickly and the control of the drone will be difficult. Therefore, the drone will not remain stable.

Proportional algorithm is not enough to eliminate the error. So, the second component which is integral control is used and it takes into account the error history [63][64]. The integral algorithm provides to reach desired position of the UAV. However, the time to reach the desired position increases. This algorithm is especially useful for eliminating negative impacts on the drone such as turbulence, irregular wind, center of gravity disorder, etc.

When the $K_i$ coefficient is increased, a softer and less aggressive flight characteristic will be obtained. High Ki coefficient also causes vibration, but its frequency is lower than high Kp coefficient.

When the $K_i$ coefficient is decreased, pitch angle of the drone may be increased while changing the direction of the vehicle, and hence, the front side of the drone will be tilted up.

The third and last component is derivative control and it acts based on the rate of error [63][64]. The derivative parameter provides future error estimation. This algorithm absorbs the shocks and it prevents "overshoot" at the beginning of takeoff. In other word, it provides smoother behavior.

When the $K_d$ coefficient is low, the movement of drone will be very smooth.

When the $K_d$ coefficient is high, the movement of drone will be very aggressive.

### 3.2.3 Communication Protocol

MavLink (The Micro Air Vehicle Link) is one of the most popular communication protocols for Unmanned Aerial Vehicles. It is widely used for real-time communication. MAVLink protocol was used for communication with UAV and on-board computer. This protocol also can be used between UAVs and GCS (Ground Control Station) over a wireless channel [65][66][67]

### 3.2.4 Ground Control Station

The ground control station is a software application that runs on a computer on the ground. This packet software typically communicates with UAV via wireless telemetry. Some of these telemetry parameters such as the angles of roll, pitch and yaw, altitude, horizontal and vertical speed, battery status, fault warnings, etc. are shown in GCS. All displayed data are real-time while operating with UAV. A Ground control station is also used to control the UAV during flight, load new mission commands, and set parameters. The GCS provides ease of monitoring in a 2D map. The other advantage of GCS is providing calibration ability of the drone [68]. The ground station software provides a “virtual cockpit” that shows many of the information. It can often be used to watch live video streams from a UAV's camera. Autonomous flight plans of the UAVs may be designed and uploaded to UAVs with the help of GCS [35].

For this project, QGroundControl Station was preferred and screen view of the control station can be seen in Figure 21.

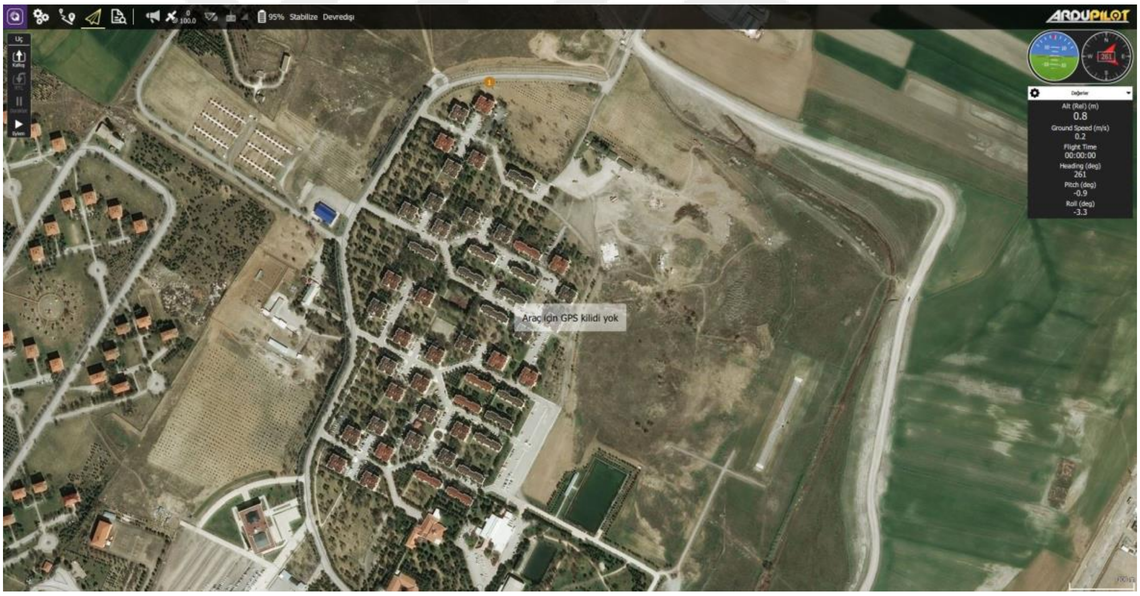


**Figure 21: QGroundControl screen view**

# CHAPTER IV

# DESIGN AND MISSION STEPS

The mission can be analyzed in four sub-sections to achieve autonomous 3D tracking task successfully.

## *4.1 Target Detection*

When the red point target is detected by the UAV, the UAV remains over the target. The resolution of the camera was set as 640 x 480 pixels in this case. This resolution values show that the horizontal area of the camera will be divided into 640 equal parts (pixels) and the vertical area will be divided into 480 equal parts (pixels). These horizontal and vertical areas are related to angle of view of the camera. Illustration of the angle of view can be seen from Figure 22.

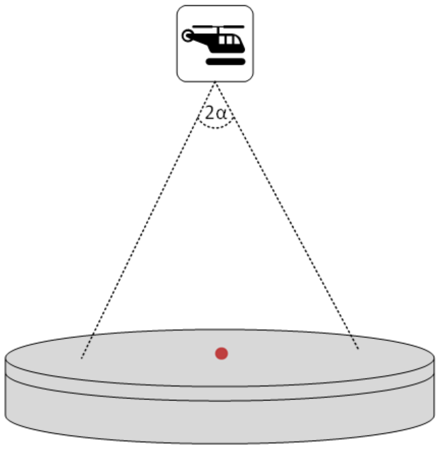


**Figure 22: Illustration of angle of view**

The angle of view values of the used camera are as follows [69]:

Horizontal viewing angle: $2\alpha_h = 53.50^{\circ}$

Vertical viewing angle: $2\alpha_v = 41.41^{\circ}$

The field of view of the camera is calculated using angles of view of the camera with the help of cosine theorem. Illustration of the field of view is shown in Figure 23 and the equations between Eq. 4 and Eq. 7 can be used for these calculations.

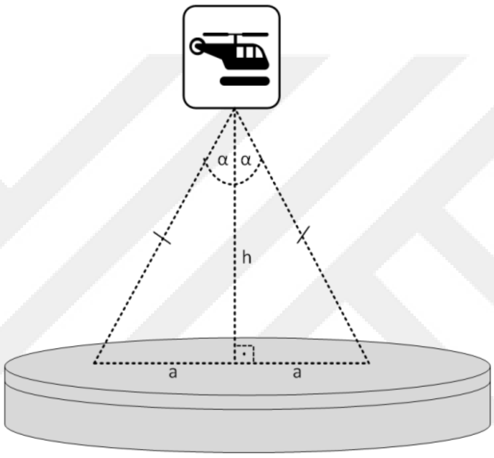


**Figure 23: Illustration of field of view**

$$\frac{h}{\mathrm{Cos}(\alpha)} = \frac{a}{\mathrm{Sin}(\alpha)} \qquad (4)$$

$$a = h \ \frac{\mathrm{Sin}(\alpha)}{\mathrm{Cos}(\alpha)} \qquad (5)$$

where,

α: Half of angle of view

a: Half of field of view

$$\mathrm{FoV_h} = 2\mathrm{h}\ \frac{\mathrm{Sin}(\alpha_\mathrm{h})}{\mathrm{Cos}(\alpha_\mathrm{h})} \qquad (6)$$

$$\mathrm{FoV_v} = 2\mathrm{h}\ \frac{\mathrm{Sin}(\alpha_\mathrm{v})}{\mathrm{Cos}(\alpha_\mathrm{v})} \qquad (7)$$

where,

$\mathrm{FoV_h}$: Field of view on horizontal axis

$\mathrm{FoV_v}$: Field of view on vertical axis

$\alpha_h$: Half of horizontal viewing angle

$\alpha_v$: Half of vertical viewing angle

In above equations, h (altitude) is measured by the barometer and this data is taken from barometer output. The calculated field of view is divided by 640 for the horizontal axis and the length corresponding to a single pixel on the horizontal axis (x-axis) is obtained. Similarly, the calculated field of view of the vertical axis is divided by 480 and the length corresponding to a pixel on the vertical axis (y axis) is calculated. These calculations can be formulated as shown in Eq. 8 and Eq. 9

$$\mathrm{P_h} = \frac{\mathrm{FoV_h}}{640} \qquad (8)$$

$$\mathrm{P_v} = \frac{\mathrm{FoV_v}}{480} \qquad (9)$$

where,

$P_h$: Unit pixel length on horizontal axis

$P_v$: Unit pixel length on vertical axis

## *4.2 Calculation of New Coordinate of Moving Target*

After target detection, it is desired that the target stays at the center of the field of view. But, to stay stable when negligible movements of the target are occurred or to prevent the oscillation of the system because of small vibrations of the UAV, a free zone was described and it was let the target move inside this described zone. The boundaries of the zone are related to application and these values were chosen as [160, 480] for horizontal axis and [120, 360] for vertical axis for this application which is equal to quarter area of the total area of the field of view. This range was described by slicing the four sides (left, right, up, down) of the center to 2 equal parts. The intersection points in all quarter parts specify the corner points of the tolerant frame. Herewith, centers of the free zone and the field of view are superimposed. The free zone frame is shown with red color in Figure 24.

If the detected target moves out of the free zone, the autonomous system follows the target and gets the target position back to the center of the field of view. In order to do this, first, the new location of the detected target is determined. By comparing the pixel location at the center of the target with the pixel location at the center of the free zone frame, it is obtained that the target is located which side of the center frame. Then the distance of the target from the center is calculated in meters by multiplying the pixel distance with pixel length.

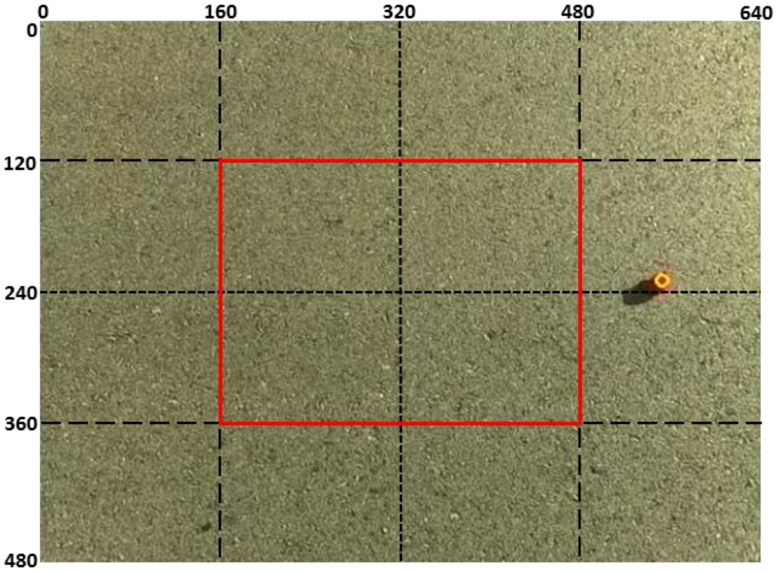


**Figure 24: Illustration of target pixel position**

The center coordinate of the field of view is (320, 240) pixels on horizontal (x) and vertical (y) axes, respectively in Figure 24. Target was detected at the pixel position of (546, 228) and it can be seen from command window in Figure 25 as well. Since the vertical 228$^{th}$ pixel is inside the tolerant frame, movement on the vertical axis is not required. However, the horizontal 546$^{th}$ pixel is out of tolerant frame, so the UAV should move to the right direction on the horizontal axis to make the target placed in the tolerance frame. General formulation to determine the required movement distance to get the target into center of the frame can be shown as between Eq. 10 and Eq. 13.

$$Pd_h = 320 - x \quad (10)$$

$$Pd_v = 240 - y \quad (11)$$

$$M_h = Pd_h \times P_h \quad (12)$$

$$M_v = Pd_v \times P_v \quad (13)$$

where,

$Pd_h$: Pixel distance from center on horizontal axis

$Pd_v$: Pixel distance from center on vertical axis

$M_h$: Required movement distance on horizontal axis

$M_v$: Required movement distance on vertical axis

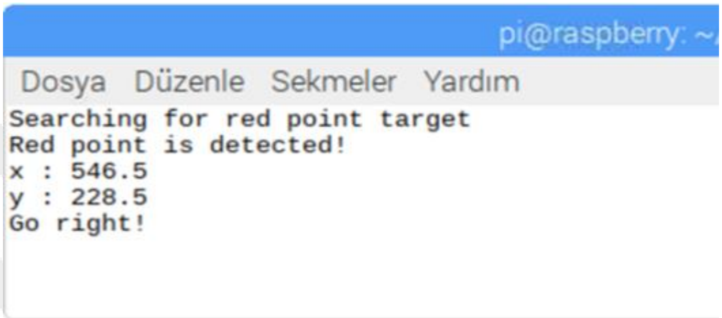


**Figure 25: Command output window for detection of target pixel position**

Calculated distance data is converted to coordinate system parameters by using coordinate calculation method. Latitude angle and Longitude angle are used as coordinate parameters. Latitude angle is changed while moving over longitude and longitude angle is changed while moving over latitude. These angles can be seen from Figure 26. By using Eq. 14 and Eq. 15, the angle differences can be calculated [70]. Obtained angle difference values are then added to current angle values to obtain the new coordinates as seen from Eq. 16 and Eq. 17.

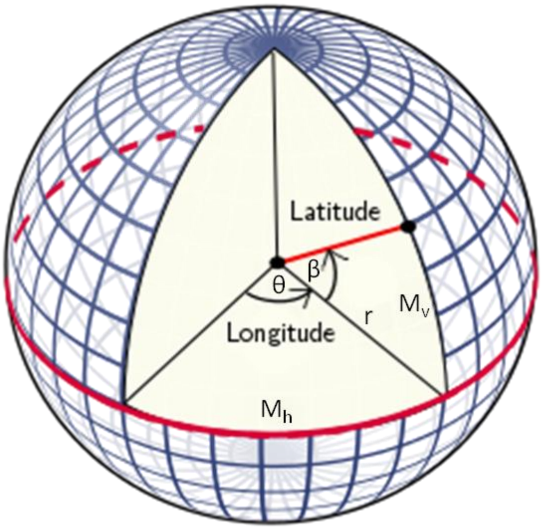


**Figure 26: Latitude and longitude angles [71]**

$$\beta = \frac{M_v}{r} \tag{14}$$

$$\theta = \frac{M_h}{r\,\mathrm{Cos}\beta} \tag{15}$$

$$\text{New Latitude angle} = \beta_{cur} + \beta \tag{16}$$

$$\text{New Longitude angle} = \theta_{cur} + \theta \tag{17}$$

where,

$\beta$: Latitude angle difference (radian)

$\theta$: Longitude angle difference (radian)

$\beta_{cur}$: Current latitude angle (radian)

$\theta_{cur}$: Current longitude angle (radian)

$M_h$: Movement distance over latitude (meter)

$M_v$: Movement distance over longitude (meter)

r: radius of the earth (meter)

## *4.3 Moving to New Coordinate*

The calculated new coordinate values are sent to the UAV via the MAVLink protocol and the UAV is made to move over the target. This part can be seen from "Results" section clearly.

## *4.4 Altitude Tracking*

After the x and y axes were checked and the position of the target was brought to the center of the frame by the autonomous UAV, the third axis, z axis (height), is checked. Then, the current distance value is compared with the target distance and the UAV behaves according to the comparison result and reaches the required height. Two different techniques have been implemented within the scope of the study.

### 4.4.1 Distance Calculation with Algorithm

In the camera-based distance calculation technique, single camera and a simple pixel algorithm are used. No other components are necessary to obtain the approximate distance. In this technique, a change in the radius of the object detected as a target is observed. The current altitude and radius are recorded when the target was first detected. If the target moves up or down while the UAV is hovering at a constant altitude after detection, the measured radius will also change. This change will be positive if the object moves upwards and approaches the UAV, and if the object moves downwards and away from the UAV the change will be negative. After the trigonometric calculations, the new radius and the previous radius are compared to determine the amount and direction of change in distance. Calculation can be made by using triangle similarity theorems. This process can be understood with the help of Figure 27, Eq. 18 and Eq. 19.

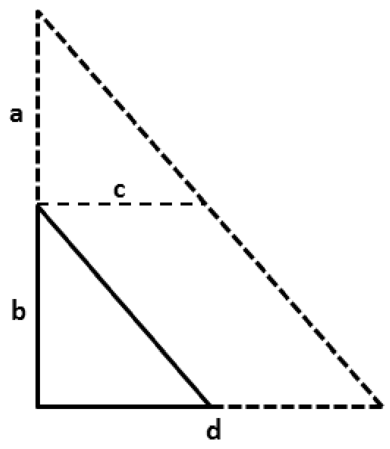


**Figure 27: Illustration of triangle similarity**

$$\frac{a}{a+b} = \frac{c}{d} \tag{18}$$

Here, "d" represents the radius of the object ($r_{ex}$) and "a+b" represents the altitude whereas "c" represents the length of the new radius ($r_{new}$) and "a" value represents the new altitude.

$$\text{ratio} = \frac{r_{new}}{r_{ex}} \tag{19}$$

If ratio > 1;

it is understood that the object is rising and approaching the UAV. The distance is calculated by multiplying the current height by calculated ratio. The obtained altitude value is sent to the UAV as an ascent command.

If ratio < 1,

it is understood that the object is descending and moving away from UAV. The distance is calculated by multiplying the result of the division by the current height. The obtained altitude value is sent to the UAV as a descend command.

There is a possible problem while calculating the distance with this technique. In some cases, even if the position of the object does not change, its radius can be measured smaller than it appears. As a result, since the autonomous UAV perceive that the object is farther from the UAV, the UAV may want to approach the object. The illustration of this relation can be seen from Figure 28.

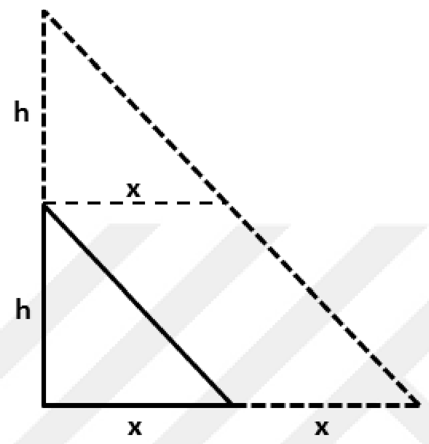


**Figure 28: Illustration of radius measurement**

When the altitude of the UAV is "2h" and the radius of the target is "2x", if the radius of the object measured as "x" instead of "2x", the UAV wants to decrease the altitude with the same ratio of radius. So, the UAV will move downwards until altitude "h". This type of system behavior is undesirable. The source of this problem may be some factors such as image quality, camera resolution, variation of light intensity or change of light angle. Some examples of this kind of problem can be seen in Figure 29 through Figure 32.

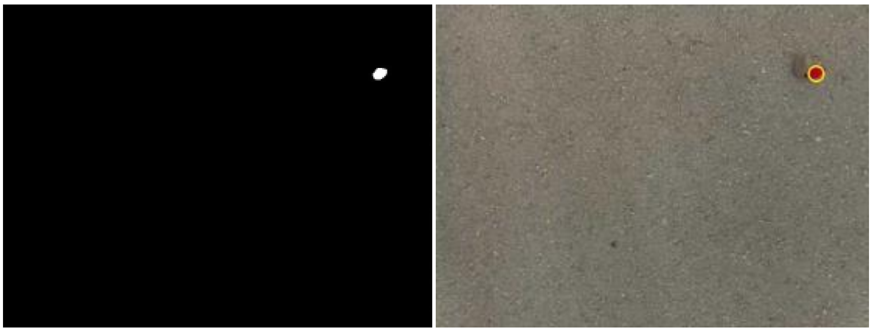

**Figure 29: Correct radius measurement**

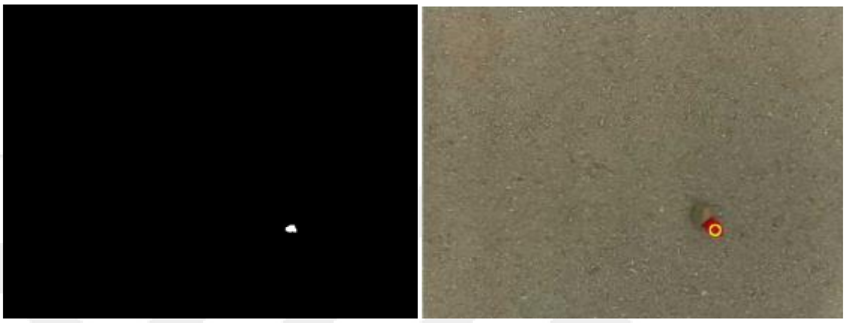

**Figure 30: Incorrect radius measurement**

**Figure 31: Correct radius measurement**

**Figure 32: Incorrect radius measurement**

In this experiment, it was observed that the probability of obtaining an incorrect value as a result of the distance measurement made by the algorithm method is higher than the measurement method made while utilizing a distance sensor. Therefore, although the distance calculation method with the algorithm is lower in cost, lighter in weight and simpler in terms of coding, it is not preferred and the mission is carried out by employing a distance sensor.

### 4.4.2 Distance Calculation with Distance Sensor

Different distance sensors can be used in this method. Two commonly used distance sensors are LIDAR and Ultrasonic sensors.

#### *4.4.2.1 Lidar sensor*

Laser technology is used in LIDAR (Light Detection and Ranging) measurement sensor [72]. The single-beam narrow-band laser is generated from the sensor and then it is reflected from target object [73]. The illustration of the working principle of Lidar sensor is shown in Figure 33. The distance can be calculated by the half of duration when the laser pulse being emitted and the time when the reflected laser pulse being sensed. The distance can be calculated by multiplying the half of the total travel duration by the speed of light (299,792,458 m/s). The result will be in meters and the formula can be seen from Eq. 20. [74].

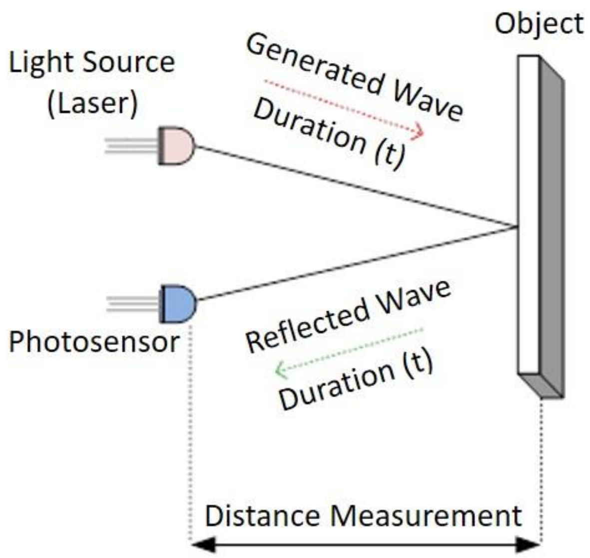


**Figure 33: Working principle of lidar sensor**

$$D = t \cdot c \qquad (20)$$

where,

D: Distance from object (m)

t: Half of time interval between generated and reflected laser pulse

c: speed of light (299,792,458 m/s)

#### *4.4.2.2 Ultrasonic sensor*

Sound wave technology is used for this type of distance sensor. The sensor generates 40 kHz sound pulse when it is triggered then waits and listens to reflected sound wave. Therefore, the total travel time (2t) of the sound wave can be measured as transmitted and reflected duration [75]. The illustration of the working principle of ultrasonic sensor is shown in Figure 34. The distance can be calculated by multiplying the half of the total travel duration (t) by the speed of the sound wave (343 m/s) [76]. The result will be in meters and the formula can be seen from Eq. 21.

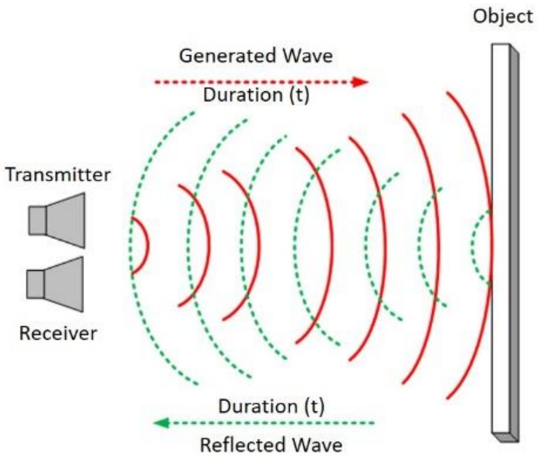


**Figure 34: Working principle of ultrasonic sensor**

$$D = t \cdot 34300 \qquad (21)$$

where,

D: Distance from the object (centimeter)

t: Half of time interval between generated and reflected sound wave (second)

From these two options, using LIDAR is high in terms of cost, and hence, it was preferred to use an ultrasonic sensor in the scope of the study.

The ultrasonic sensor is located very close point to the lens of the camera used for image processing. The sensor thus calculates the distance corresponding to the center of the field of view. When the red point object is first detected during the task, the distance between the UAV and the red object is recorded as "target distance". A hysteresis loop-style control mechanism was integrated to algorithm to prevent the oscillation because of minor measurement error and small swings of the drone. In this context, a tolerance of 10% was allowed for distance change. That is, if the object is between (0.9 x target distance) and (1.1 x target distance), the UAV will continue to maintain its current position. If the height of the target changes more than the specified tolerance, the result obtained by sensor measurement is recorded as a "current distance" and this value is compared with the first recorded value which is target distance. If the current measured value is smaller than the target distance, it is understood that the object rises and approaches the UAV. The new desired altitude value is calculated by adding the obtained difference value with the current altitude. The calculated altitude value is sent to the UAV as an ascent command and the drone rises up. If the current distance is greater than the target distance, it is understood that the object is lowered away from the UAV. The new altitude is calculated by subtracting the obtained difference from the current altitude. The calculated altitude value is sent to the UAV as a descend command and the drone descents.

This process can be represented with the following equations; Eq. 22 and Eq. 23. Altitude tracking photos can be seen from Figure 35 to Figure 37.

$$\mathrm{D} = \mathrm{C} - \mathrm{T} \tag{22}$$

$$\mathrm{N} = \mathrm{A} - \mathrm{D} \tag{23}$$

where,

A: Current Altitude (obtained from barometer)

C: Current Distance. The current distance between UAV and target (obtained from ultrasonic sensor)

D: Difference from target distance

N: New Altitude

T: Target Distance. The distance between UAV and target at first target detection (constant, obtained from ultrasonic sensor)

**Figure 35: Detection of the target**

**Figure 36: Altitude tracking of the target**

**Figure 37: Measurement of displacement in z-axis (height)**

# CHAPTER V

# RESULTS

A simple test mission is illustrated in Figure 38 and tracking commands are given as shown in Figure 39. The command window output was simplified to reduce the amount of writing and enhance the readability. The basic functional behavior of the autonomous UAV system can be observed from the test track shown in Figure 38. The UAV was taken off from Home position (represented by H letter with green color). While searching the field, a red point target was detected at the point shown as "Tracking is Starting" with yellow color. Then the related commands were sent to the flight controller by on-board computer according to movement of the target. Movement positions were drawn by ground control station with purple line and directions were shown with red arrows. Commands were added on the figure with red color as well. At point "A", the target was placed 1 meter above the ground and this difference was detected by UAV as seen from the command window. Finally, target was placed in the previous position again and the UAV was moved down until it reached target distance as seen in the command window. At this point, the mission was complete, and the point was marked as "End Point" and indicated by the yellow arrow. The photo of the autonomous 3D target tracking mission in action from “Mission Planner GCS” is shown in Figure 40.

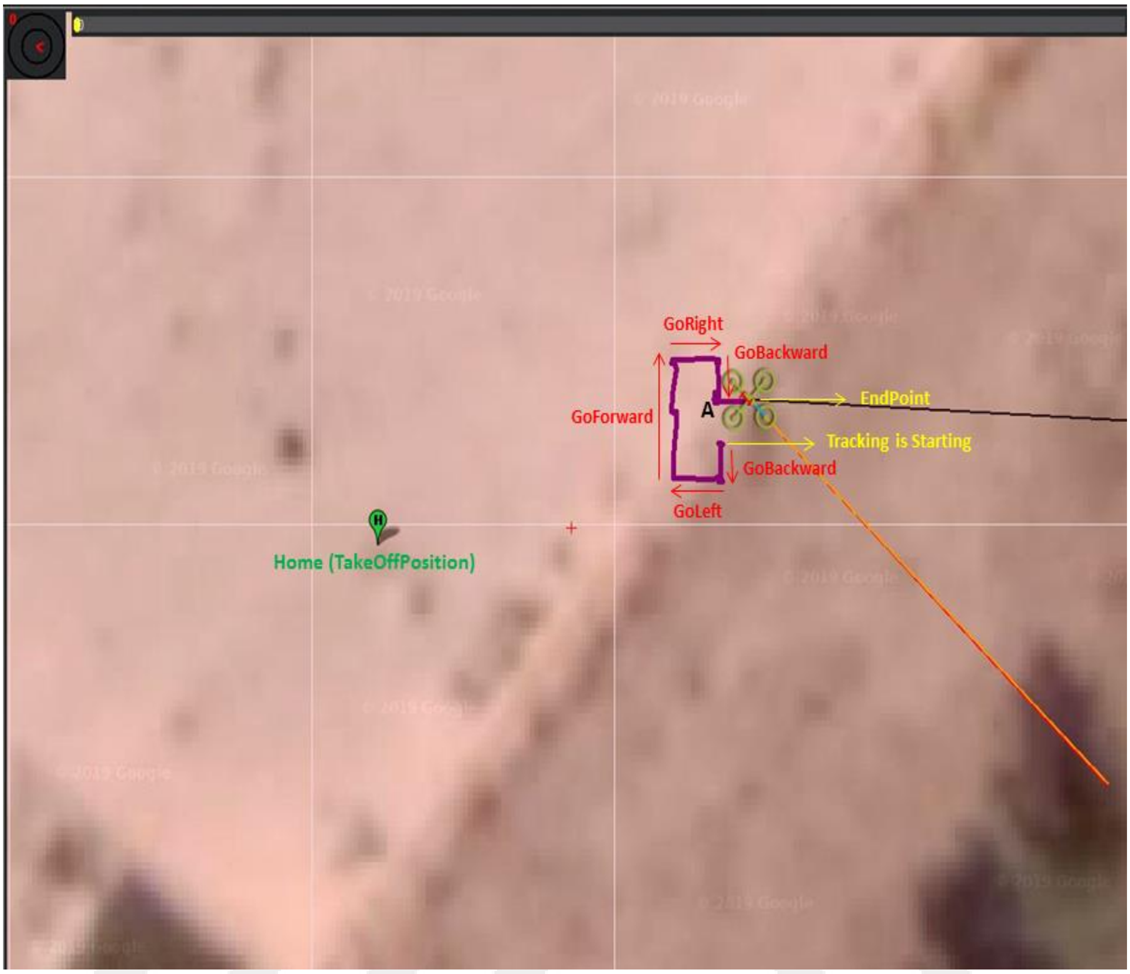


**Figure 38: Illustration of 3D tracking mission**

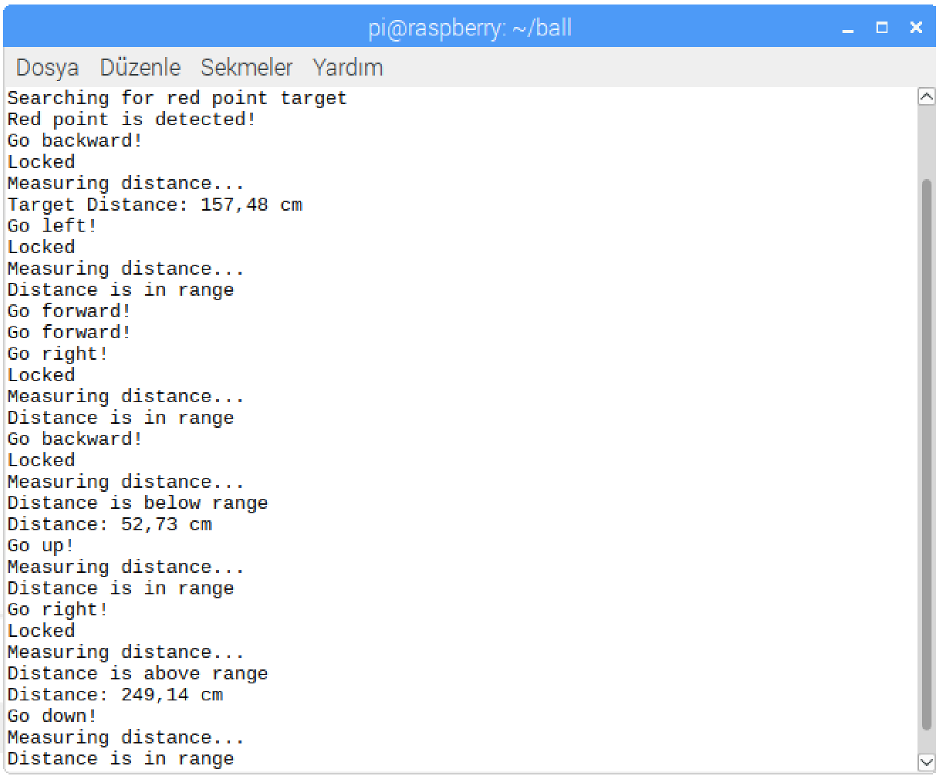


**Figure 39: Command output window of 3D tracking mission**

**Figure 40: Photo of 3D tracking mission**

# CHAPTER VI

# CONCLUSION

A target object can be detected and can be tracked on all three axes (x, y, z axes) with the aid of the UAV, image processing system and a distance sensor. The distances to be moved for right-left (x axis/roll) and forward-backward directions (y axis/pitch) were calculated by using coordinate calculation methods, subsequently they are sent to flight controller via MavLink protocol and the UAV reached the target positions successfully. It has been shown that using a single camera and basic distance algorithm option for tracking altitude (z axis) is not a good enough solution to get stable results due to environmental factors and systemic offsets in the camera sensor. The mission that three-dimensional (x, y, z axes) tracking of the object defined as a target was successfully completed using a single camera, distance sensor and hence a simpler algorithm by taking into account factors such as cost, weight and ease of application. Such a system can be customized for various tracking applications. The system can be improved significantly by using long-range Lidar sensors and high-resolution cameras for applications where the cost is not as important concern. Thus, tracking tasks can be performed more accurately and precisely with such an upgraded system.

# BIBLIOGRAPHY


[1] R. Barták and A. Vyškovský, "Any Object Tracking and Following by a Flying Drone", Fourteenth Mexican International Conference on Artificial Intelligence, Cuernavaca, Mexican, Oct. 2015.

[2] D. E. Knuth, *The TEX Book*. Reading, Massachusetts: Addison-Wesley, 1984. Reprinted as Vol. A of *Computers & Typesetting*, 1986.

[3] D. E. Knuth, “The WEB system for structured documentation, version 2.3,” Tech. Rep. STAN-CS-83-980, Computer Science Department, Stanford University, Stanford, California, Sept. 1983.

[4] Baykar, Bir Başarı Hikayesi: BAYKAR. Retrived March 14, 2020, from https://www.baykarsavunma.com/haber-Bir-Basari-Hikayesi-BAYKAR.html

[5] C. Karaağaç, İHA Sistemleri Yol Haritası, STM, 2016.

[6] M. Arjomandi, S. Agostino, M. Mammone, M. Nelson M and T. Zhou, "Classification of Unmanned Aerial Vehicles", MECH ENG 3016 Aeronautical Engineering, The University of Adelaide Australia, 2006.

[7] M.R. Bhatt, "Solar Power Unmanned Aerial Vehicle: High Altitude Long Endurance Applications", The Faculty of the Department of Mechanical and Aerospace Engineering, San Jose State University, California, USA, 2012.

[8] T. Breverton, Breverton's first world war curiosities. USA: Amberley, 2014.

[9] V. Prisacariu, B. Herciu and A. Luchian, "Aspects Regarding the Electric Propulsion of the UAV Multicopter", Review of the Air Force Academy, Brasov, Romania, Dec. 2017.

[10] S.J. Zaloga, Unmanned Aerial Vehicles. Oxford, UK: Osprey, 2008.

[11] R.K. Barnhart, S.B. Hottman, D.M. Marshall and E. Shappee, Introduction to Unmanned Aircraft System. Boca Raton, USA: CRC Press, 2012.

[12] G. Cai, K. Lum, B.M. Chen and T.H. Lee, "A brief Overview on Miniature Fixed-Wing Unmanned Aerial Vehicles", 8th IEEE International Conference on Control and Automation, Xiamen, China, Jun. 2010.

[13] J.D. Rae, Analyzing the Drone Debates. New York, USA: Palgrave macmillan, 2014.

[14] P. B. Charlesworth, "A Game Theoretic Approach to Coordinating Unmanned Aerial Vehicles with Communications Payloads", Cardiff University, UK, Jul. 2015.

[15] T.J. Mueller, "On the Birth of Micro Air Vehicles", Internal Journal of Micro Air Vehicles, vol. 1, pp. 1-12, Mar. 2009.

[16] R.J. Hansman and R. Weibel, "Human and Automation Integration Considerations for UAV Systems", MIT International Center for Air Transportation Conference, USA, Sep. 2004.

[17] M. Franklin, "Unmanned Combat Air Vehicles: Opportunities for the Guided Weapons Industry", Royal United Services Institute for Defence and Security Studies, UK, Sep. 2008.

[18] C. Dang, H. Pham, T. Pham and N. Truong, "Vision Based Ground Object Tracking Using AR.Drone Quadrotor", International Conference on Control, Automation and Information Sciences (ICCAIS), Nha Trang, Vietnam, Nov. 2013.

[19] J. Pestana, J.L. Sanchez-Lopez, P. Campoy and S. Saripalli, "Vision Based GPS-denied Object Tracking and Following for Unmanned Aerial Vehicles", Linkoping, Sweden, Oct. 2013.

[20] R. Barták, A. Hraško and D. Obdržálek, "A Controller for Autonomous Landing of AR.Drone ", The 26th Chinese Control and Decision Conference (CCDC), Changsha, China, May. 2014.

[21] J. Junell, E. Kampen, C. Visser and Q. Chu, "Reinforcement Learning Applied to a Quadrotor Guidance Law in Autonomous Flight", AIAA Guidance, Navigation, and Control Conference, Florida, USA, Jan. 2015.

[22] A. Mashood, A. Dirir, M. Hussein, H. Noura and F. Awwad, "Quadrotor Object Tracking using Real-Time Motion Sensing", 5th International Conference on Electronic Devices, Systems, and Applications, Ras Al Khaima, United Arab Emirates, Dec. 2016.

[23] A. Chakrabarty, R. Morris, X. Bouyssounouse and R. Hunt, "Autonomous Indoor Object Tracking with the Parrot AR.Drone", International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA USA, Jun. 2016.

[24] D. A. Mercado-Ravell, P. Castillo and R. Lozano, "Visual Detection and Tracking with UAVs, Following a Mobile Object", Advanced Robotics, vol.33, pp. 1-15, 2019.

[25] L. Feng and Q. Fangchao, "Research on the Hardware Structure Characteristics and EKF Filtering Algorithm of the Autopilot PIXHAWK", Sixth International Conference on Instrumentation & Measurement, Computer, Communication and Control, Harbin, China, Jul. 2016.

[26] G. Zhang and L. Hsu, "Intelligent GNSS/INS Integrated Navigation System for a Commercial UAV Flight Control System", Aerospace Science and Technology, vol.80, pp. 368-380, Sep. 2018.

[27] W. Saengphet, S. Tantrairatn, C. Thumtae and J. Srisertpol, "Implementation of System Identification and Flight Control System for UAV", 2017 3rd International Conference on Control, Automation and Robotics, Nakhon Ratchasima, Thailand, Apr. 2017.

[28] ST, STM32F427 Datasheet, Rev. 9, Jul. 2016.

[29] ST, AN4365 Application Note, Using STM32F4 MCU Power Modes with Best Dynamic Efficiency, Rev. 2, May. 2014.

[30] Raspberry Pi, Raspberry Pi 3 Model B. Retrived April 20, 2019, from https://www.raspberrypi.org/products/raspberry-pi-3-model-b/

[31] Raspberry Pi, Camera Module. Retrived May 5, 2019, from https://www.raspberrypi.org/documentation/hardware/camera/

[32] TD Gulf, Raspberry Pi Camera. Retrived May 5, 2019, from https://www.tdgulf.com/product/raspberry-pi-camera-v1-3-5mp-1080p/

[33] Ublox, Neo-6M GPS Modules Datasheet, Rev. E, Dec. 2011.

[34] J. Connors, "On the subject of Solar Vehicles and Benefits of the Technology", Proceedings of the International Conference on Clean Electrical Power ICCE'07, Capri, Italy, May. 2007.

[35] R. Benveniste, B. Gökçe, Radyo Kontrol ve Yer İstasyonları Tasarımı, 2018 Tübitak Uluslararası İnsansız Hava Araçları Yarışması, Apr. 2018.

[36] Roboshop, Pixhawk PX4. Retrived January 10, 2020, from https://www.roboshop.com.tr/pixhawk-full-seti?search=pixhawk

[37] Robokits, 433 Mhz Telemetry Module Pair for Pixhawk and APM 100 mW 2 km Range, Retrived January 23, 2020, from https://robokits.co.in/drones-quad-hexa-octa-fpv/fpv-video-telemetry-osd/433mhz-telemetry-module-pair-for-pixhawk-and-apm-100mw-2km-range

[38] AltiGator Unmanned Solutions. Retrived July 1, 2020, from https://drones.altigator.com/radio-telemetry-kit-a-for-pixhawk-433mhz-p-42334.html

[39] C. Yuchao and L. Hui, "Study on the Meso-Structure Image of Shale Based on the Digital Image Processing Technique", International Conference on Image Analysis and Signal Processing, Taizhou, China, Apr. 2009.

[40] L. Xiao-hong, "Development and Application of Image Processing Laboratory System Based on DSP", Journal of Hefei University, vol. 19, pp. 38-41, Apr. 2009.

[41] O. Marques, Practical Image and Video Processing Using MATLAB. New Jersey, USA: Wiley, 2011.

[42] M. Lutz, Learning Python. California, USA: O'Reilly Media, 2007.

[43] A.F.J. López, M.C.P. Pelayo and A.R. Forero, "Teaching Image Processing in Engineering Using Python", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, USA, Jun. 2017.

[44] A. Bogdanchikov, M. Zhaparov and R. Suliyev, "Python to Learn Programming", Journal of Physics: Conference Series, vol. 423, pp. 1-5, 2013.

[45] W. Saengphet, S. Tantrairatn, C. Thumtae and J. Srisertpol, "Implementation of Mobile Platform Using QT and Open CV for Image Processing Applications", 3rd International Conference on Control, Automation and Robotics, Nakhon Ratchasima, Thailand, Apr. 2017.

[46] S. Brahmbhatt, Practical OpenCV. USA: Apress, 2013.

[47] R. Laganière, OpenCV Computer Vision Application Programming Cookbook Second Edition. Birmingham, UK: Packt, 2014.

[48] X. Jia, "Fabric Defect Detection Based on Open Source Computer Vision Library OpenCV", 2nd International Conference on Signal Processing Systems (ICSPS), Jiaxing, China, Jul. 2017.

[49] R. Chityala and S. Pudipeddi, Image Processing and Acquisition using Python. Florida, USA: CRC Press, 2014.

[50] OpenCV-Python Tutorials, Changing Colorspaces. Retrived July 8, 2019, from https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_colorspaces/py_colorspaces.html

[51] OpenCV, Thresholding Operations using inRange. Retrived July 23, 2019, from https://docs.opencv.org/3.4/da/d97/tutorial_threshold_inRange.html

[52] OpenCV, Smoothing Images. Retrived July 10, 2019, from https://docs.opencv.org/3.4.2/d4/d13/tutorial_py_filtering.html

[53] A. Koschan and M. Abidi, Digital Color Image Processing. New Jersey, USA: John Wiley & Sons, Inc, 2008.

[54] B. Jähne, Digital Image Processing. Berlin, Germany: Springer, 2005.

[55] W. K. Pratt, Digital Image Processing. New Jersey, USA: John Wiley&Sons, Inc, 2001.

[56] OpenCV, Morphological Transformations. Retrived August 13, 2019, from https://docs.opencv.org/3.4.2/d9/d61/tutorial_py_morphological_ops.html

[57] R. M. Haralick and L.G. Shapiro, Computer and Robot Vision. Boston, USA: Wesley, 1992.

[58] K.A.M. Said, A.B. Jambek and N. Sulaiman, "A Study of Image Processing Using Morphological Opening and Closing Processes", Internal Journal of Control Theory and Applications, vol. 9 (31), pp. 15-21, Jan. 2016.

[59] Morphological Image Processing. Retrived May 1, 2020, from https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm

[60] OpenCV, Eroding and Dilating. Retrived August 13, 2019, from https://docs.opencv.org/master/db/df6/tutorial_erosion_dilatation.html

[61] G. Sudha and S. N. Deepa, "Optimization for PID Control Parameters on Pitch Control of Aircraft Dynamics Based on Tuning Methods", Applied Mathematics & Information Sciences, pp. 343-350, Jan. 2016.

[62] İ. Coşkun and H. Terzioğlu, "Hız Performans Eğrisi Kullanılarak Kazanç (PID) Parametrelerinin Belirlenmesi", Journal of Technical Online, vol. 6, pp. 180-205, 2007.

[63] R. Gamasu and V.R.B. Jasti, "Robust Cohen-Coon PID Controller for Flexibility of Double Link Manipulator", International Journal of Control and Automation, vol. 7, pp. 357-368, 2014.

[64] K. J. Åström and T. Hägglund, PID Controllers: Theory, Design, and Tuning. NC, USA: Instrument Society of America, 1995.

[65] S. Atoev, K. Kwon, S. Lee and K. Moon, "Data Analysis of the MAVLink Communication Protocol", International Conference on Information Science and Communications Technologies, Tashkent, Uzbekistan, Nov. 2017.

[66] A. Allouch, O. Cheikhrouhou, A. Koubâa, M. Khalgui and T. Abbes, "MAVSec: Securing the MAVLink Protocol for Ardupilot/PX4 Unmanned Aerial Systems", 15th International Wireless Communications & Mobile Computing Conference, Tangier, Morocco, Jun. 2019.

[67] A. Koubâa, A. Allouch, M. Alajlan, Y. Javed, A. Belghith and M. Khalgui, "Micro Air Vehicle Link (MAVlink) in a Nutshell: A Survey", IEEE Access, vol.7, pp. 87658-87680, Jun. 2019.

[68] J. C. del Arco, D. Alejo, B. C. Arrue, J.A. Cobano, G. Heredia and A. Ollero, "Multi-UAV Ground Control Station for Gliding Aircraft", 23rd Mediterranean Conference on Control and Automation (MED), Torremolinos, Spain, Jun. 2015.

[69] Raspberry pi. Camera Module. Retrived May 14, 2019, from https://www.raspberrypi.org/documentation/hardware/camera/

[70] J. Case, Astro Navigation Demystified. Weymouth, UK: Bookcase Learning Resources, 2016.

[71] Latitude and Longitude are angles, not just some numbers on the earth's surface. Retrived August 15, 2019, from https://medium.com/@atinders/latitude-and-longitude-are-angles-not-just-some-numbers-on-the-earths-surface-d16019aaf24a

[72] J. Laconte, S. Deschênes, M. Labussière and F. Pomerleau, "Lidar Measurement Bias Estimation via Return Waveform Modelling in a Context of 3D Mapping", International Conference on Robotics and Automation, Montreal, Canada, May 2019.

[73] X. Li and Y. Guo, "Application of LiDAR Technology in Power Line Inspection", International Conference on Advanced Materials, Intelligent Manufacturing and Automation, China, May 2018.

[74] G. Duh, Light Detection and Ranging (LiDAR), Lecture notes. Portland State University, Portland, USA.

[75] N. A. Latha, B.R. Murthy and K.B. Kumar, "Distance Sensing with Ultrasonic Sensor and Arduino", International Journal of Advance Research, Ideas and Innovations in Technology, vol. 2, pp, 2016.

[76] A. Pantano and D. Cerniglia, "Simulation of Laser-generated Ultrasonic Wave Propagation in Solid Media and Air with Application to NDE", Applied Physics A, vol. 98, pp. 327-336, 2016.

# VITA

Şeyma Sivritaş was born in 1991. She received the B.Sc. degree of Electrical and Electronics Engineering from Dokuz Eylül University, İzmir, Turkey in 2015. She worked at Vestel Electronics between 2015 and 2018. She has been working at Turkish Aerospace as an Avionics System Engineer. She has experience in aircraft avionics systems design, power distribution system design and analysis of communication systems.