

# Autonomous Navigation of Unmanned Aerial Vehicles based on Android Smartphone

Talal Bonny<sup>1</sup>, Mohamed B. Abdelsalam<sup>2</sup>  
Department of Electrical and Computer Engineering  
College of Engineering, University of Sharjah, UAE

**Abstract**—In the past few years, the adoption of drone technology across industries has increased dramatically as more businesses started to recognize its potential, uses, and scale of global reach. This paper proposes a design solution of a smart GPS quadcopter aircraft navigation system, discusses its hardware and software implementation process and eventually, analyses and reports the final test results. The flight path of the quadrotor is remotely manipulated via an Android based Graphical User Interface. This outdoor handheld application allows the operator to select a point of interest through Google map satellite view; consequently, the quadrotor takes off then hovers and ultimately lands on the destination location. Instructions in conjunction with coordinates are sent and received throughout a web server which serves the communication operation between the smartphone and the quadrotor. Experimental results yield fruitful data communication and successful autonomous flight control with smooth and stable maneuvering.

**Keywords**—UAV; drone; quadrotor; android; Arduino; GPS; GPRS; GSM

## I. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) are deemed one of the utmost pioneering aviation innovations in the history of humankind. In fact, the first attempts traced back to the early 1990s. The French aircraft designer Louis Charles Breguet was the first to build a rotary wing aircraft, in 1907, which was able to lift itself off the ground [1]. Few years later, in 1920, Oehmichen tried six other rotorcrafts designs [2]. The second design he built, which uses four rotors and eight propellers, demonstrated considerable degree of stability for its time and remained airborne for several minutes. More robust prototypes emerged in 1922, 1956 and 1958 [3], [4]. Ever since, endeavors continued to improve and advance, thus, sophisticated yet simple designs and refined prototypes proliferated and; consequently, scholars are, presently, racing to develop competitive applications of quadrotors in various sectors and multiple domains. From surveillance, reconnaissance and weapon delivery, in military and law agencies, through Internet of Vehicles [5], to aerial imagery and television news coverage [6], [7], [8]. Besides, it is widely utilized in commercial transportation of goods such as food, pharmaceutical products and parcels delivery [8]. To process the data, a controller usually is used such as CPU [9], [10], FPGA [11], [12], or Microcontroller as explained in this work. Drones and control systems play an important role in our daily life, from hobbies like photography to more important uses like surveillance and military purposes. All these applications focus on specific usage, all of which lack a navigation system to move the drones autonomously from one location to another. Any application that will use drones will have to call them or send them to different

locations. Our objective is to provide a friendly prototype that provides the utilities which will execute these operations. Lots of contemporary research studies attempted at designing and developing different approaches and various methodologies towards the goal of achieving autonomous aerial navigation maneuvers [13]. In this context, [14] proposed a framework, operational in unknown indoor and outdoor environments, of a Micro Aerial Vehicle (MAV) with autonomous capabilities using one monocular camera, as exteroceptive sensor, along with an onboard Inertial Measurement Unit (IMU). Likewise, the work presented by [15] extended the earlier research study to implement an innovative approach of automatic navigation systems using two cameras with non-overlap in their respective field of views (FOVs). Similarly, and according to [16], the navigation of aerial vehicles can be accomplished using multi-sensor data fusion method which determines precise pose parameters of the vehicle in real time. In another work, [17] used information obtained from an RGB-D camera to successfully develop an autonomous flight control system. Besides, [18] proposed a navigation framework that enables small-scale UAVs to navigate autonomously in foliage environment using a 2D laser range finder. The framework contains real-time onboard motion estimation and trajectory smoothing. Despite their accuracy and precision, these solutions use complex algorithms in order to operate and result in heavyweight platforms which make them inconvenient for daily usage. Consequently, other lightweight solutions were explored; hence, autonomous navigation of aerial vehicles throughout onboard Android smartphones were recently employed and developed [13], [19], [20], [21]. These communication devices are equipped with sensors and IMUs essential for estimating flight condition data and orientation of the quadrotor. Moreover, they have computational power (processor) useful in implementing flight control algorithms. Some approaches used supplementary computer besides the smartphone, as in [19], where it used wireless networks to communicate with a PC. Using this connection, data could be transmitted from the smartphone to the PC, for instance live images or log data. It is also used to send commands from the PC to the quadcopter, for example angle references. Reference [13], unlike other works, proposed a different approach in which the smartphone has the control, guidance and navigation systems implemented inside, depending on the information provided by the onboard sensors. Twitter account was used to enable the drone to share information and receive commands in multi-agent system. The solution in this paper proposes a novel methodology of achieving autonomous aerial navigation operations of a quadrotor with very simple, lightweight, customizable and inexpensive design framework using a combination of onboard GPS/IMU

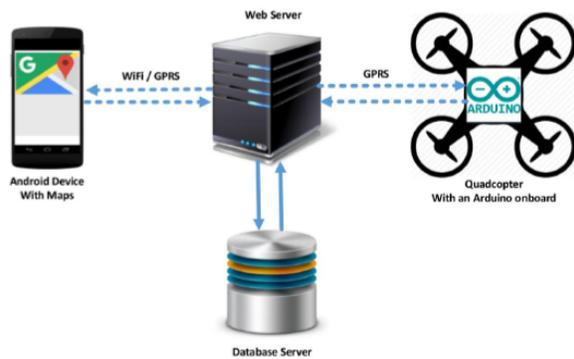


Fig. 1: System Overview

and GSM/GPRS technology. Furthermore, the proposed design allows the autonomous control of a quadcopter using the developed application. Fig. 1 illustrates the design of the prototype proposed. It consists of two main components; the quadcopter and the smartphone. The latter enables the operator to select the waypoint he wishes the drone to hover to. This is accomplished via a developed android-based application which uses Google Maps APIs. Commands are sent to a web server via Wi-Fi. The received data will be stored in a database, for future usage. The web server ensures the bidirectional communication between the two ends. As a second phase, the server forwards the data to the Arduino microcontroller mounted on the drone. The latter responds back with the requested information using a GPRS, and the server, in its turn, directs it to the android device. The design criteria considered during the implementation of the proposed application are: flight durability and stability, ease of implementation and maintenance, usability, customizability and cost effectiveness with long wireless communication range. The paper is organized as follows. Section II illustrates the technical background. Section III discusses the building methodology of the quadrotor along with a step-by-step process of achieving a successful navigation operation. Section IV presents the implementation and testing of the software module. Section V shows the Communication Operation. Section VI introduces the results and the analysis. Constraints are listed in Section VII. Section VIII conducts the conclusion.

## II. TECHNICAL BACKGROUND

This section discusses some of the important technical concepts in which this project was based upon.

### A. Drones

Unmanned Aerial Vehicles (UAV), commonly known as drones are aircrafts without a human pilot aboard. Controlling UAVs flight can be achieved in different ways such as using autonomous control, using onboard computers or by using a remote control. Remotely controlled UAV are the most common type of drones, but in the last few years autonomous UAV applications and functionalities increased especially after the introduction of flight controllers. A Flight Controller

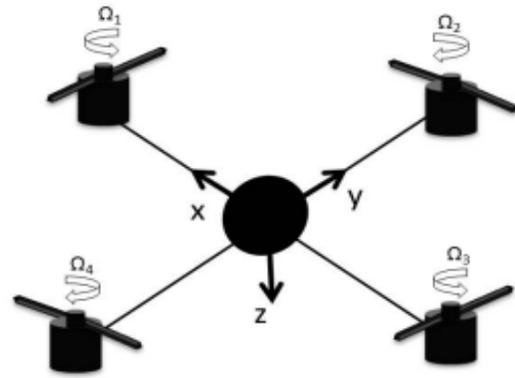


Fig. 2: Quadrotor configuration

(FC) is a small circuit board of varying complexity. Its function is to direct the Round per Minute (RPM) of each motor in response to an input. A command from the pilot of the multi-rotor is fed into the flight controller, which determines how to manipulate the motors accordingly<sup>1</sup>. Most of the flight controllers employ sensors to supplement their calculations. These sensors could be a Gyroscope for orientation, a Barometer to hold altitudes, or a GPS for auto-pilot and fail-safe purposes. The first drone ever created was an autonomous torpedo that drops and explodes at a preset time, made by Dayton-Wright Airplane Company during the World War 1. As of late 1980's, usage of drones expanded after they have been proven efficient in surveillance and monitoring purposes. Nowadays, drones' popularity is increasing rapidly. In civil applications, drones can be used in many fields such as policing, surveillance and monitoring, rescuing, and inspection. Cinematography is one of the most common applications of drones nowadays. Companies like DJI and Parrot manufacture special types of multi-rotors drones that can take very high-quality videos and pictures. Furthermore, drones nowadays are being used in delivery fields. Multi-rotors are compact aircraft that contains more than two rotors. This type of drones is the most common between civilians due to its low price, customizability, ease of use and small size. Some of the multi-rotor aircraft are the Tricopter which has 3 motors, a Quadcopter with 4 motors and Hexacopter with 6 motors. Their main advantage compared to other drones is that it can be easily assembled and used in different fields like cinematography, surveillance, and search operations. In the past years, many countries have adapted to this technology and made some rules and regulation regarding flying multi-rotors.

**Quadrotor flight dynamics:** Many literature surveys addressed the topic of quadrotor flight dynamics [22]. This nonlinear dynamic model is necessary for flight stability and for accurate control. The quadrotor has four motors, each two diagonal rotors spin in the same direction and against the direction of the adjacent rotors [23], as shown in Fig. 2.

The model has six degrees of freedom, thus six distinct variables express its position in the space. The variables  $[x, y$

<sup>1</sup>[https://en.wikipedia.org/wiki/Unmanned\\_aerial\\_vehicle](https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle)

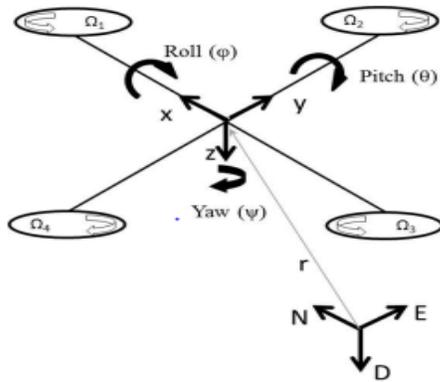


Fig. 3: Euler angles in quadrotors

, z] represent the distances of the quadrotor's center of mass along the x, y, and z axes, respectively. Whereas, the variables  $[\Phi, \Theta, \Psi]$  denote the orientation of the quadrotor. They are known as Roll, Pitch, and Yaw, respectively [23], [24]. Fig. 3 clearly explains the Euler angles.

The most common type of multi-rotors is the Quadcopter or "Quad". The Quad has four motors, each two diagonal rotors spin in the same direction and against the direction of the rotors next to them. When all the rotors spin in equal magnitudes in addition to the airfoil blades lift is generated, and due to the symmetrical and unique shape of the Quad thrust is achieved by the Roll and Pitch. The Roll and Pitch can be achieved by increasing the rotational speed of two neighbor rotors. As they spin in deferent directions they negate the reaction of the rotating blades, and because their lift is higher than the other two rotors they cause the Quad to tilt applying some force sideways and creating the thrust. When increasing the speed of two diagonal rotors the reaction from rotating the blades in a certain direction increases, leading to the increase in the reaction of the blades on the Quad, making the Quad rotate to the opposite direction thus creating the Yaw. Fig. 4 demonstrates how to apply the flight mechanisms on a Quadcopter.

### B. GSM/GPRS

Global System for Mobile communications (GSM) is an open, digital cellular technology used for transmitting mobile voice and data services<sup>2</sup>. It uses the frequencies 900 MHz and 1.8 GHz in Europe, whereas in the US it uses 850 MHz and 1.9 GHz. GSM supports voice calls, transmission of Short Message Service (SMS), and data transfer speed up to 9.6 kbps. GSM was designed to be a secure wireless system which relied on user authentication using a pre-shared key, challenge-response, and over-the-air encryption. General Packet Radio Service (GPRS), sometimes referred to as 2.5G, is used by GSM for data transmission, it offers a throughput rate up to 40 kbps, enabling mobiles to access online services in a reasonable speed at a rate similar to a dial-up modem. GPRS is a best-effort service, which means that the throughput and latency that depend on the number of other users sharing the

service concurrently. GPRS supports the Internet protocol (IP), and the Point-to-point protocol (PPP).

## III. BUILDING METHODOLOGY AND NAVIGATION OPERATION

### A. Building Methodology

1) *Hardware components:* Now-a-days, plenty of companies are producing affordable easy to fly and ready to use drone models. Nevertheless, a DIY approach of building a quadrotor is more favorable as it guarantees the hardware customizability; it facilitates regular upgrades if desired and further enhancements when needed. Fig. 5 demonstrates the general structure of the quadcopter constructed. The main mechanical components necessary for building the drone are:

**Body frame:** The DJI F450 body frame kit was used. It consists of four arms and an average sized center plate made of carbon fiber, as shown in Fig. 6. Lately, due to its structural rigidity and light weight, the demand for carbon fiber composites have increased. The frame holds all the components together through the arms and the center plate. The latter is usually reserved for both flight controller and battery. Its robustness and resistance to crashes are the main reasons behind its selection.

**Propellers:** Two types of propellers exist; carbon fiber propellers, and plastic propellers. Due to their stiffness, carbon fiber propellers produce less vibrations and a lower propeller spinning noise; however, they are twice as expensive as the regular plastic propeller. Quadrotors use two clockwise (CW) and two counter-clockwise (CCW) propellers. While designing a quadrotor, various factors should be considered; flight usage, payload of the quadrotor and the KV. KV is the number of revolutions per minute that the motor will turn when 1 V is applied. Large propellers are mainly used in two fields, the cinematography and weight carrying. On the other hand, smaller propellers are mainly used in drone racing. In the proposed design, the 10x45 large plastic propellers were used due to the stability they provide to the drone in addition to their low price and proven efficiency.

**Electric motors:** Thrust is the main aspect when it comes to choosing motors. Thrust needed per motor can be calculated theoretically by the following equation:

$$thrust = \frac{\text{total weight} \times 2}{\# \text{ motors}}$$

The estimated total weight of this mechanical design is calculated as follows:

$$\text{battery (450 g)} + \text{APM2.8 flight controller (20 g)} + \text{body frame (280 g)} + \text{motor (50} \times \text{4g)} \quad (1)$$

Hence, the minimum thrust required per motor is equivalent to  $(11302)/4 = 565$  g thrust.

As a result, the DJI 2212 - 920KV brushless motor was used. Placing the motors and the propellers at equal distances, provides both, best performance and simplest control algorithms.

The electrical components required in quadrotors are:

<sup>2</sup><http://www.gsm.com/aboutus/gsm-technology/gsm>

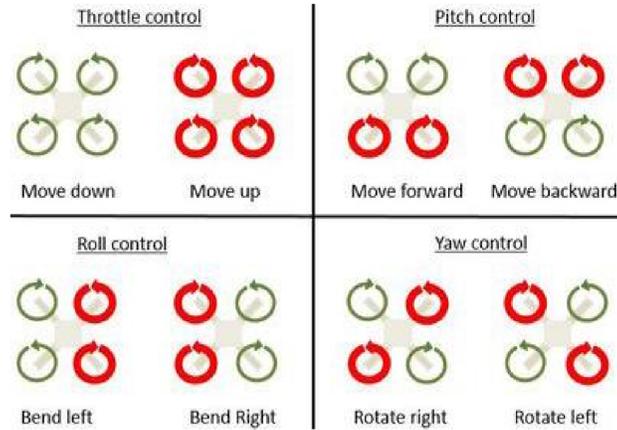


Fig. 4: Flight mechanisms of a Quadcopter

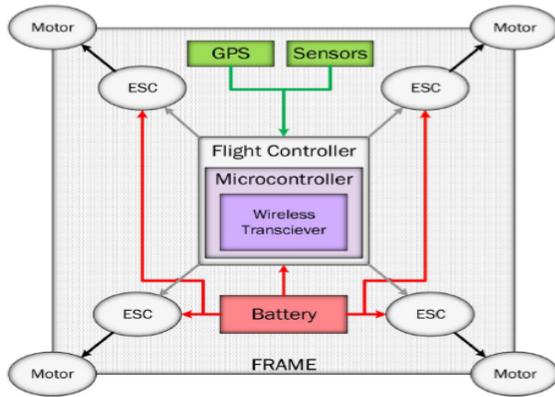


Fig. 5: Quadrotor mechanical design



Fig. 7: Live data of APM through Mission Planner



Fig. 6: DJI F450 Body Frame

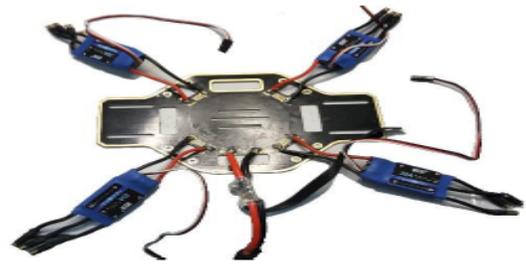


Fig. 8: Lower plate of the drone

**Electronic Speed Control (ESC):** The role of an ESC is to control and regulate the speed of a motor. In this design, it must be able to provide a maximum current of 30 A. The Simonk 30A ESC was used thanks to its reliability and low cost. Quadcopters use four motors each is connected to one ESC. Flight controllers send signals to the ESC, which drive the brushless motor. Battery Elimination Circuit (BEC) is usually built within the ESC, which is a voltage regulator to convert the Li-Po battery voltage (2S-7.4V, 3S-11.1V, 4S-14.8, etc.) to a smaller voltage like 5V.

**Controller board:** It contains different controllers

**Flight controller:** ArduPilot Mega (APM) 2.8 flight controller was used for a number of reasons. In fact, it is programmable, easy to use, fair price and has all the sensors required to attain a stable flight. In order to program it, a software called Mission Planner (AP) was used. Mission planner provides the ability to get live data midflight, as shown in Fig. 7. Flight controller continuously reads inputs from sensors, and changes the speed of the motors based on these inputs.

**Microcontroller:** The Arduino Uno microcontroller was used due to its low price, low power consumption, and simplicity.

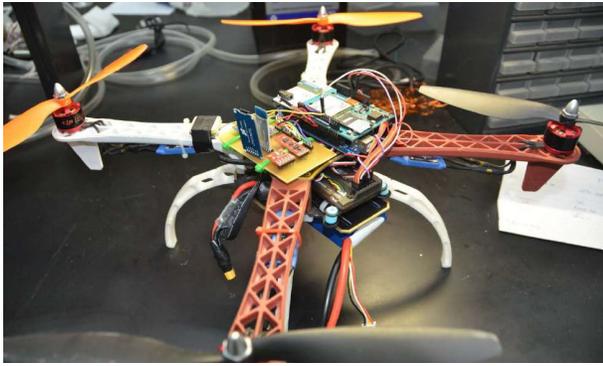


Fig. 9: The Arduino connected to the HC05 on the drone with APM on top

ity with regard to programming. Furthermore, its clock speed fits the developed prototype. Moreover, it supports different modules of sensors, actuators, and transceivers, in addition to shields. Shields are circuits which can go on top of the Arduino Uno. They can provide different utilities from FM transceivers to Wi-Fi modems.

**Battery:** It is the only power supply for the designed quadrotor unless an ESC without built-in BEC was used. In such systems, another 5V power supply is needed to supply the flight board and the motors. LiPo batteries consist of cells, each cell has a nominal voltage value of 3.7V. For example, a 4S LiPo battery indicates that we have four cells in series providing a voltage of  $4 \times 3.7 = 14.8$  V. The HRB 3S LiPo battery, with a capacity of 6000 mAh, was used because it satisfies the design requirements in terms of weight and flight duration in addition to its reasonable price.

2) *Components assembly and testing:* Component testing approach was performed. Upon positive functionality of each and every unit, integration phase was next; hardware parts were combined together incrementally. Fig. 8 shows the ultimate structure of the lower plate of the quadrotor after hardware assembly. The power module (voltage regulator) is connected to the battery which powers the flight controller and the ESCs. The actual battery is placed between the upper and lower body plates. The flight controller is connected to the rest of the components in the upper plate, as Fig. 9 illustrates. Afterwards, the flight of the quadrotor built was tested with a remote control. The flights were very stable with smooth maneuvering.

3) *Testing a functional quadrotor:* A ready to fly drone was used in order to get hands on knowledge on flight mechanisms and a thorough understanding on the behavior of the remote control in conjunction with the sort of responses it sends to the flight controller.

Initially, the Remote Control (RC) receiver was connected to the oscilloscope. The responses gained from the receiver designated the start and the end of each channel used. As more and more tests were performed, more channels were connected, simultaneously, as shown in Fig. 10. Results demonstrated that two consecutive connected channels do not interfere with one another and each channel had the same period. Channels provided 1.5 ms width pulses at normal state, 1.8 ms width

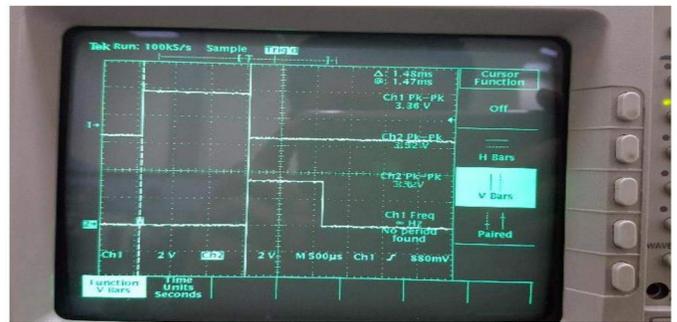


Fig. 10: RC signals

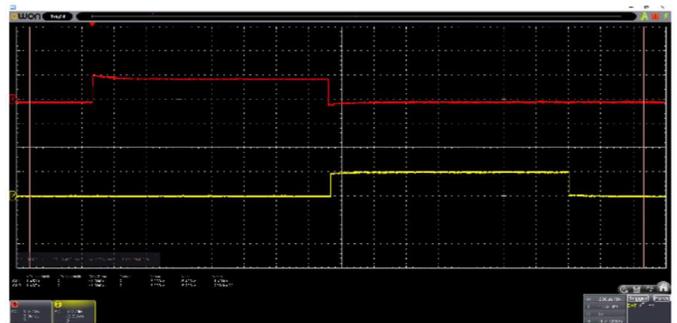


Fig. 11: Signals produced by Arduino Mega

pulses at high state and 1.2 ms width at low state. Channels could be ordered ascendingly (in terms of pulses), as follows: Channel 1, Channel 3, Channel 2, then Channels 4 - 8. Remote controls differ from one another by the number of channels they support. A channel corresponds to one command only. The RC used, in the previous test, had eight channels; four for the main movement directions and four extra channels designated for operator's usage. The other channels could be used to control mounted cameras or gimbal systems.

4) *Imitating RC receiver's behaviour:* Following the integration of all hardware parts, the constructed quadrotor was tested via a remote control. Results showed stable flights with smooth maneuvering. Next, the automation process started by imitating the behaviour obtained from the RC receiver. This was accomplished using Arduino Mega Pulse Width Modulation (PWM) pins. The signals obtained showed similar behaviour as the signals produced using the RC receiver, as can be seen in Fig. 11 below.

Testing the automation of the key operations, take-off and landing, came afterwards. This was accomplished by connecting channel #3 to the Mega and leaving all other channels connected to the RC receiver, for safety reasons. This test allowed to achieve the first semi-autonomous flight.

5) *GSM/GPRS Transceiver:* Arduino GSM Shield module enables the Arduino board to make phone calls, send/receive SMS messages and to connect to the internet over the GPRS wireless network. This is done thanks to the GSM library included in Arduino IDE. The GSM shield is used thanks to its compatibility with Arduino board and its ease of use and implementation. In order to successfully operate, the shield has

to be, simply, plugged onto the board. Similarly, a SIM card, offering a GPRS coverage, has to be plugged in.

6) *GPS installation:* In order to allocate the drone, a Global Positioning System module (GPS) has to be connected to the Arduino Mega microcontroller. For this purpose, the NEO-6M GPS module is used; a module with high accuracy and low power consumption, in chorus. Moreover, it is a basic module that is compatible with most microcontrollers. In this proposed design, the margin of error should not exceed 3 meters, this range of error is acceptable because of the size of streets and turns in the roads.

7) *IMU installation:* Knowing the direction and the orientation of the drone is essential while navigating. As a result, the Integrated Measurement Unit (IMU), manufactured by SparkFun, was one of the solutions adopted. It contains the following [25]:

- 3-axis Magnetometer sensor which provides information about its location in space and the direction of the drone.
- 3-axis Accelerometer sensor to measure the speed of the drone.
- 3-axis Gyroscope sensor which measures the tilting angle. Also, it can be used to add more stability to the quadrotor.

The use of the 3-axis magnetometer sensor is crucial as the navigation operation requires sensing angle difference across points. In order to conduct accurate navigation, it is required that the starting angle is considered the reference angle. Left and right rotations depend on the readings of the sensor. They alter the heading of the quadrotor through the Yaw angle. For instance, if the net angle is less than 180 degrees then the drone rotates to the left otherwise, it rotates to the right [25].

As an alternative for the ultrasonic sensor, the pressure sensor “barometer” was used. The barometer takes the initial location pressure as a reference and continuously measure the pressure along the journey to estimate the attitude from pressure reading. The pressure of the altitude is measured using the following equations [26]

$$h_{alt} = \left(1 - \left(\frac{P_{sia}}{1013.25}\right)^{0.190284}\right) \times 145366.45$$
$$h_m = 0.3048 \times h_{alt}$$

where,

$h_{alt}$  is the altitude measured in [ft]

$P$  is pressure measured in [mb]

$h_m$  in the altitude measured in [m]

#### IV. MOBILE APPLICATION DESIGN

##### A. Test Application 1

This application was developed in order to make sure that Google services such as Google Maps can be successfully imported and used. The application renders a map view, gets

the phone’s current location and zooms in on it. In addition, few preset markers are present which can be placed and removed by clicking the corresponding buttons. Besides, there is a slider to change the mode of the map view.

##### B. Test Application 2

This application was developed to test the Google Directions API. The application starts by rendering a map where the operator is prompted to set two markers. Afterwards, the shortest navigational path between the two points is computed and, consequently, a list of latitude and longitude coordinates is returned. Then, a polyline, representing the navigational route, is displayed as well as the list of coordinates and the total number of points alongside the path.

##### C. Final Application

The prototype is designed to work on Android device that has Android 4.1.0 (Jelly Bean) or higher, this is due the lack of some features in earlier versions of Android. The core functionality of the main application is as follows: once the mobile application is launched, the user is prompted to enter his login credentials; username and password. The input data is validated against the database and accordingly, five distinct cases could possibly emerge:

- Case 1 - “Login Successful”: the database grants the username and password combination. Hence, the user is directed immediately to a google map page.
- Case 2 - “Missing Credentials”: this occurs when the user forgets to enter either the username or the password.
- Case 3 - “Incorrect Combination”: this happens when the user enters a wrong username or password.
- Case 4 - “Connection Error”: this denotes that the database is currently inaccessible. Reasons could be an internet connection failure or a momentary server failure.
- Case 5 - “Drone Busy”: this means that the requested drone is currently in use by another user. Thus, the user is informed that he is able to select the drone only when its status changes to “available”.

Fig. 12 is a flow chart showing the sequence of the android application. Upon granted access, the operator can easily perceive the label on the map. It denotes the actual location of the quadrotor. The user picks a point of interest by clicking on the map. It is the drone’s flight destination. Consequently, a method, which is responsible of creating requests to Google’s servers, is invoked. The method considers the current location of the drone as the starting point; whereas, the selected point as the ending point.

Consequently, the server returns an array of points representing the latitude and longitude coordinates of the route which leads to the user’s point of interest. Besides, a polyline of the navigation path is drawn on the map.

The quadcopter would not fly until the user clicks on the “Start” button. Once done, the android application sends a

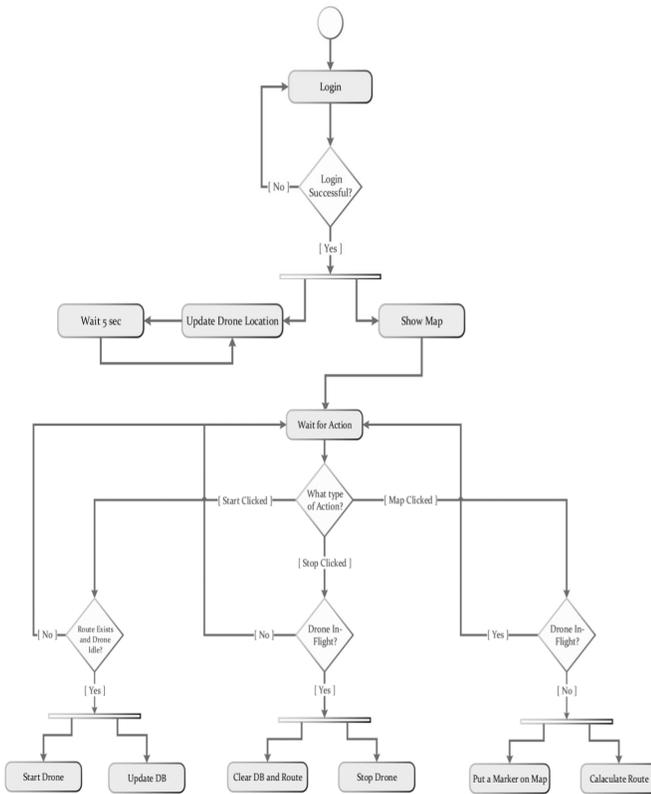


Fig. 12: Android application activity diagram

request to the database with all the required points in order for them to be saved and read later on for navigation.

The user is not allowed to request a new navigation path until the actual flight comes to an end or a stop has been issued. The stop button terminates all sorts of activities the drone is actively performing. As a result, the latter's status is set to "available", hence it can conduct new flights.

In the case where the drone is idle, for a period of time, its status alters to "available", thus other users would be able to connect to it and maneuver it. In the case where the user issues commands to a drone already in use, the database would issue a time out and the user would be directed to the home page (login page).

## V. COMMUNICATION OPERATION

Establishing a successful connectivity between the software and the hardware modules was the next milestone to reach. As mentioned earlier, the web server is the communication tool that ensures data exchange between the two parties. Participants in this operation are the:

- database
- web server
- quadrotor
- operator

Subsections below demonstrate, in details, the steps followed to design, test and implement the communication operation.

### A. Database

Fisrt, an IP address along with a domain name were reserved to the database. In addition, a server, linked to the database IP address, was reserved. Hosting24 services were used the help accelerate the process. Second, a demo database was created to test the connection between the database and the web server. Eventually, the server was able to successfully execute SQL operations like SELECT, UPDATE, and INSERT. The third step was the database design. Fig. 13 below illustrates the entity relation diagram of the database. It is a one-to-one relationship between both the drone and the smartphone entities. In fact, the smartpone navigates the drone via commands and a list of coordinates (longitudes, latitudes). Moreover, the smartphone identifies the drone through a unique ID and it is authenticated through password. The drone responds by the GPS coordinates of its actual location.

### B. Web Server

The web server receives HTTP requests, from both the smartphone and the drone. Firstly, a reserved IP address and a domain name were allocated to the server. Secondly, simple codes were developed to test the connectivity with the database and to ensure that simple SQL queries are operational. Thirdly, the different scenarios that could happen, during communication, were established and identified along with the data to be exchanged. In fact, both the drone and the Android-based application send the required data values through URL Special Characters encoding which, can be embedded in the HTTP request of the page.

The data exchanged, at the web server level, can be seen from two perspectives; smartphone perspective and quadrotor perspective.

### C. Smartphone Perspective

User log in:

In order to establish connection with the quadrotor, the user needs to be authenticated. The operator transmits, to the server, the following: drone ID, password, and a unique user identifier (i.e. MAC Address). The response will be one of the following options:

- 1) Access granted, followed by the drone current location
- 2) Invalid data / Missing credentials
- 3) Connection error has occurred
- 4) Incorrect ID/Password combination
- 5) Drone is busy
- 6) Update Error

Fig. 14 shows the activity diagram for user login.

### D. Quadrotor Perspective

The drone will transmit its current location. The response will be one of the following options:

- 1: To stop the drone
- 2 - 4: Reserved for different user commands, followed by a list of waypoints the drone has to follow)

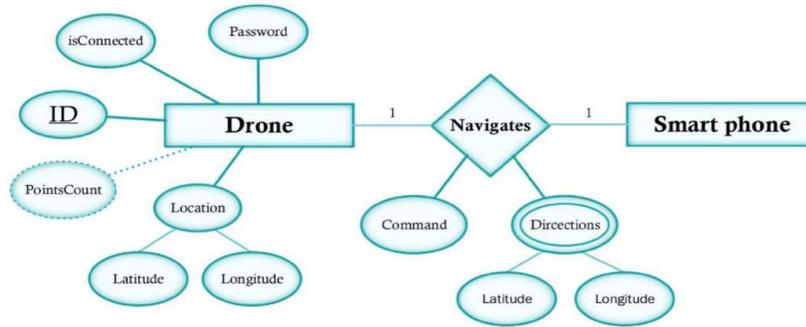


Fig. 13: ER Diagram of the Database

- 5: Connection error has occurred
- 6: Invalid transmitted data / Missing credentials

Fig. 15 presents the activity diagram of the web server.

#### E. Controlling the Drone

Once the connection with the quadrotor is established, the monitoring of the drone, by the user, is now enabled. The response will be as follows:

- 1: Update successful
- -1: Invalid data / Missing credentials
- -2: Connection error has occurred
- -3: Timeout (the user has been leaving the drone idle for a while)
- -4: Duplicate entries (already exist )
- -5: INSERT Error
- -6: UPDATE Error
- -7: Inconsistent data

The final step was the creation of web pages content. PHP language was used as it is HTML-embedded and it supports database operations. Fig. 16, below, shows the activity diagram.

#### F. Drone (Arduino)

In order to develop the communication system, at Arduino level, the first step was to make sure that the Arduino is connected to the service provider. The simplest way to test the GSM connectivity is by sending and receiving an SMS message. The second step was to get connected to GPRS services. In order to do this, three main attributes were required:

- service provider Access Point Name (APN)
- login
- password

The last step was to communicate with the web server by sending the current location of the quadrotor, and receiving the commands and the direction.

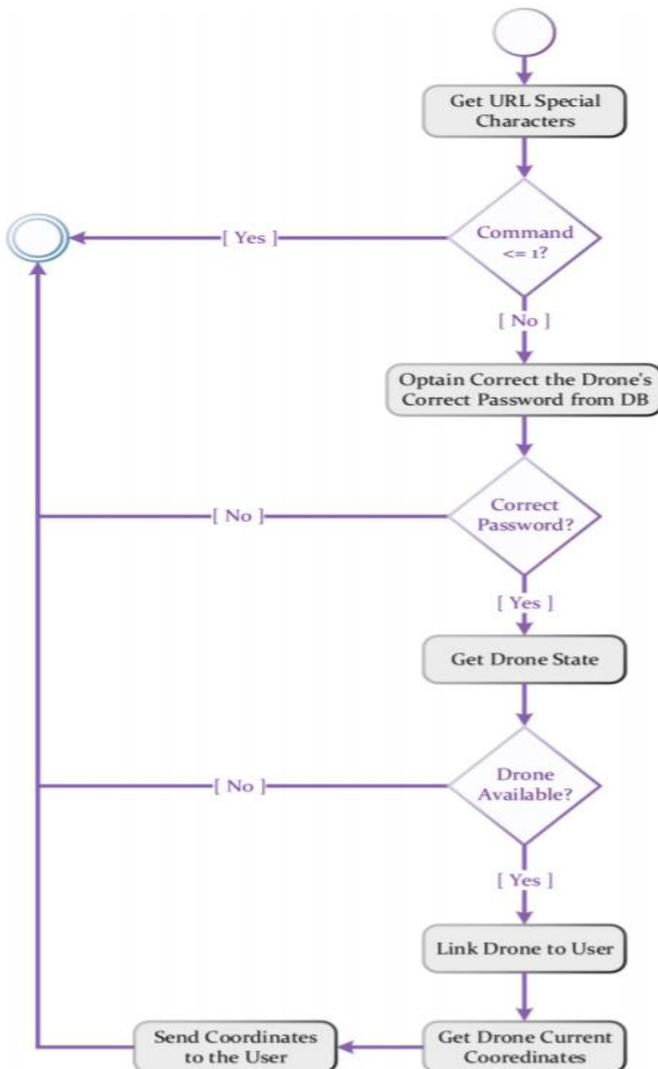


Fig. 14: Activity diagram for user login

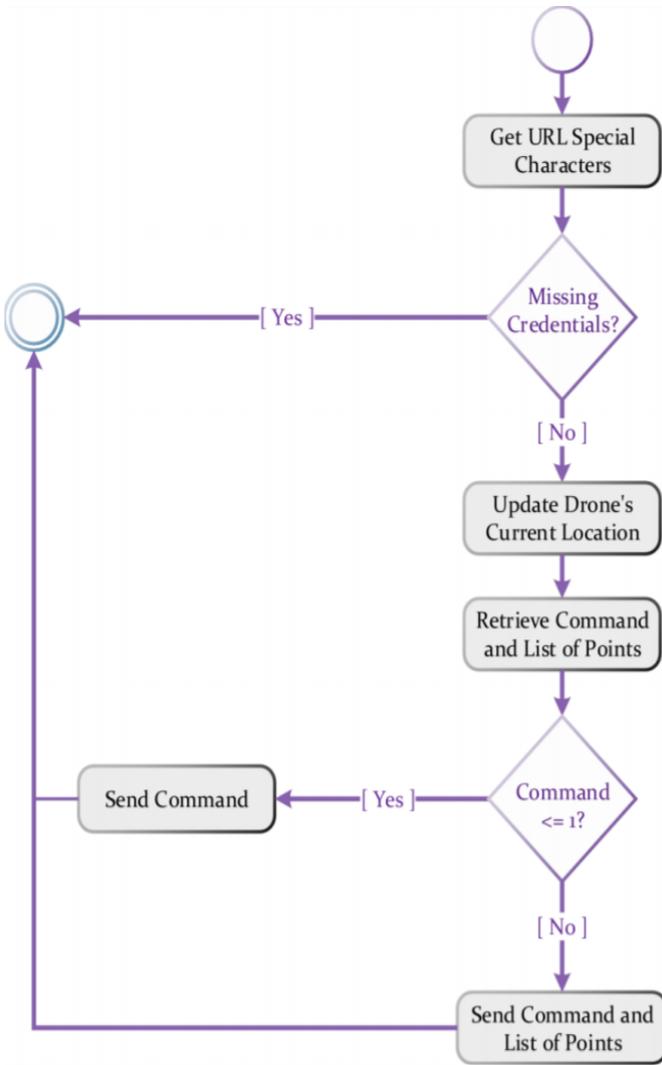


Fig. 15: Drone.php activity diagram

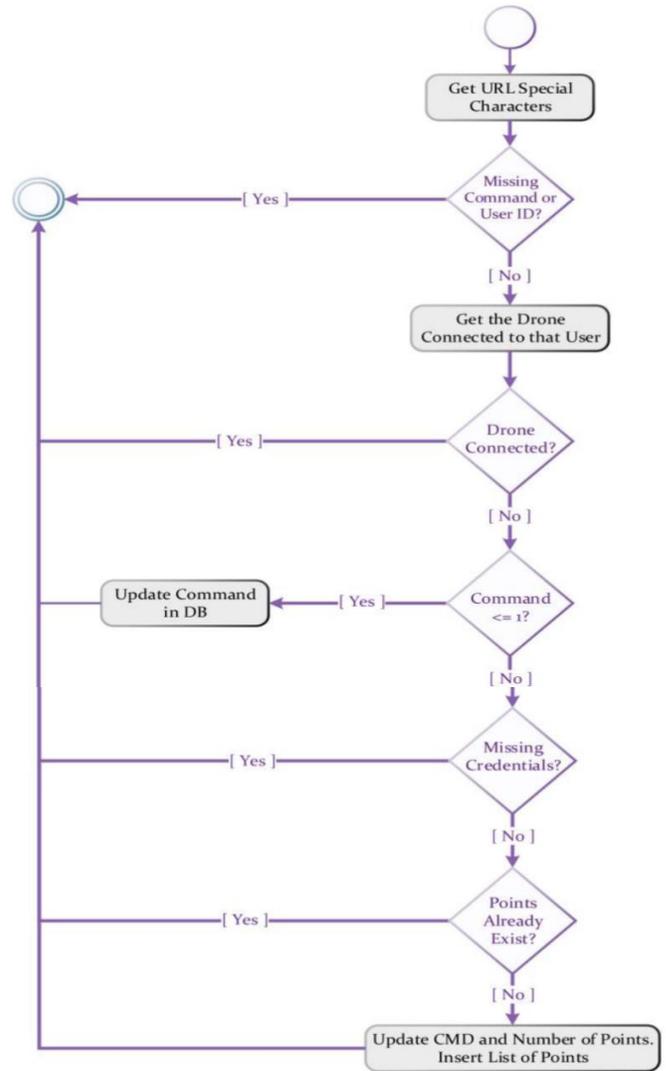


Fig. 16: CMD.php activity diagram

### G. User (Android)

In order to develop and establish the communication between the android and the server, the first step was to obtain permissions to access internet from the android device. Second, was testing the HTTP libraries followed by testing the connection with the server.

### H. Overall System

At a final stage, the prototype was assembled together and tested as a whole.

## VI. RESULTS AND ANALYSIS

Experiments and analysis demonstrated that the proposed prototype is fruitfully accomplished. In fact, data transmission between android application and Arduino board is successful and consequently, the quadcopter is effectively monitored via the smartphone app. While the prototype is fully functional and meets the design criteria set, it has some limitations and shortcomings which can be considered in further developments. In

fact, the quadrotor uses a battery as power supply and will not be operational once the battery is dead. Hence, users will need to recharge the battery before each use of the drone. Besides, the flight time of the quadcopter is limited by the payload lifted and the battery size. Furthermore, any sort of failure in the web server will cut off the communication between the drone and the smart phone. Moreover, the GPS reading is not accurate in tight streets which affects the efficiency of the navigation. In addition, the design is fragile against specific weather conditions such as rain, wind, snow, etc. And it does not avoid obstacles. Also, the drone has to be directed to the north upon starting. Likewise, quality of communication is proportional to the quality of the service provider. Similarly, the coverage area of the communications depends on the coverage range allowed by the service provider.

## VII. LIST OF CONSTRAINTS

Some of the limitations of this design would be that the Quadcopter runs on a battery and will not work once the charge of the battery runs out. Users will have to recharge the

battery before they can use it again. Also, the Quadcopter's flight duration is restricted by the battery size and the weight it carries. The quality of the communication depends on the quality of the service provider and the coverage area of the communications depends on the service provider. The Quadcopter must not fly over a No-Fly Zones like airports or military facilities, which constrains the Quad to fly over streets only. The communication between the Quadcopter and the smart phone is done through a Web server, and any fault in the server will cut off the communication between them. For the GPS, its accuracy could make some problems in tight streets, affecting the efficiency of the navigation. Bad weather conditions affect the Quadcopter usage.

### VIII. CONCLUSION

In summary, autonomous drones are gaining more and more attention by highbrows and researchers. As a result, their usage will only grow with time. This article attempted at presenting the design and implementation of a simple, customizable and cost effective smart solution of a remotely monitored GPS quadrotor through an Android-based mobile platform. The user needs to be successfully logged-in in order to establish connection with the quadrotor and start maneuvering operations. Consequently, controlling the drone is enabled for the operator. In addition, the quadrotor sends information and receives instructions through the GSM/GPRS module. This work can be employed eminently in multiple and diverse quadrotor applications as it should open the door to improved flight control and autonomous hover in the near future.

### REFERENCES

- [1] J. G. Leishman, "The breguet-richet quad-rotor helicopter of 1907," *National Academies of Sciences, Engineering, and Medicine*, vol. 47, pp. 58–60, 2001.
- [2] S. Spooner, "A successful french helicopter," *Flight*, vol. XVI, p. 47, 1924.
- [3] "A successful french helicopter," *Flight*, p. 247, 1960.
- [4] I. H. Culver, T. F. Hanson, and L. G. Look, "Helicopter rotor system," Patent US3 261 407A, 08 05, 1964.
- [5] M. Bassem and T. Bonny, "Iov road safety: Vehicle speed limiting system," *IEEE International Conference On Communications, Signal Processing, and Their Applications*, 2019.
- [6] M. Gao, X. Xu, Y. Klinger, J. van der Woerd, and P. Tapponnier, "High-resolution mapping based on an unmanned aerial vehicle (uav) to capture paleoseismic offsets along the altyn-tagh fault, china," 2016.
- [7] M. Fuhrmann and M. C. Horowitz, "Droning on: Explaining the proliferation of unmanned aerial vehicles," 2017.
- [8] R. Luppincini and A. So, "A technoethical review of commercial drone use in the context of governance, ethics, and privacy," 2016.
- [9] M. A. Z. Talal Bonny and K. N. Salama, "An adaptive hybrid multiprocessor technique for bioinformatics sequence alignment," *The 5th Cairo International Conference on Biomedical Engineering Conference, (CIBEC'10)*, 2010.
- [10] A. Allagui, A. E. Rojas, T. Bonny, A. Elwakil, and M. A. A. Kareem, "Nonlinear time-series analysis of current signal in cathodic contact glow discharge electrolysis," *Journal of Applied Physics*, 2016.
- [11] T. Bonny and A. S. Elwakil, "Fpga realizations of high speed switching-type chaotic oscillators using compact vhdl codes," *Journal of Nonlinear Dynamics*, 2018.
- [12] T. Bonny and S. Henno, "Image edge detectors under different noise levels with fpga implementations," *Journal of Circuits, Systems and Computers*, 2018.
- [13] M. Hayajneh, M. Melega, and L. Marconi, "Design of autonomous smartphone based quadrotor and implementation of navigation and guidance," *Elsevier*, 2016.
- [14] M. Achtelik, M. Achtelik, S. Weiss, and R. Siegwart, "Onboard imu and monocular vision based control for mavs in unknown in- and outdoor environments," *IEEE*, 2011.
- [15] haowu Yang, S. A. Scherer, and A. Zell, "Visual slam for autonomous mavs with dual cameras," 2014.
- [16] F. Samadzadegan and G. Abdi, "Autonomous navigation of unmanned aerial vehicles based on multi-sensor data fusion," *IEEE*, 2012.
- [17] R. G. Valenti, I. Dryanovski, C. Jaramillo, D. P. Ström, and J. Xiaoi, "Autonomous quadrotor flight using onboard rgb-d visual odometry," *IEEE*, 2014.
- [18] J. Q. Cui, S. Lai, X. Dong, P. Liu, B. M. Chen, and T. H. Leei, "Autonomous navigation of uav in forest," *IEEE*, 2014.
- [19] A. Bjalemark and H. Bergkvist, "Quadcopter control using android based sensing," *Advances in Electrical and Computer Engineering*, 2014.
- [20] G. Loianno, G. Cross, C. Qu, Y. Mulgaonkar, J. A. Hesch, and V. Kumar, "Flying smartphones: Automated flight enabled by consumer electronics," *IEEE*, 2015.
- [21] G. Loianno, Y. Mulgaonkar, C. Brunner, D. Ahuja, A. Ramanandan, M. Chari, and S. Diaz, "Smartphones power flying robots," *IEEE*, 2015.
- [22] N.-S. Pai, W.-C. Li, M.-H. Chou, and P.-Y. Chen, "Flight control for a quadrotor of attitude control based on android system and using optimal-tuning design," *elsevier*, vol. 54, pp. 170–184, 2016.
- [23] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," 2007.
- [24] S. Swarnkar, H. Parwana, M. Kothar, and A. Abhishek, "Development of flight dynamics model and control of biplane-quadrotor uav," 2018.
- [25] J. E. Kain and C. Yates, "Airborne imaging system using global positioning system (gps) and inertial measurement unit (imu) data," Patent US 005 894 323A, 03 22, 1996. [Online]. Available: <https://patents.google.com/patent/US5894323A/en>
- [26] D. Burch, *The Barometer Handbook: A Modern Look at Barometers and Applications of Barometric Pressure*, T. Burch, Ed. Starpath Publications, 2009.