UNMANNED SURFACE VEHICLE WITH ANALYTICAL DROWN MAP SYSTEM

SAAD MAZIN SHIHAB AL-DAKHEEL

September, 2019

UNMANNED SURFACE VEHICLE WITH ANALYTICAL DROWN MAP SYSTEM

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED

SCIENCES OF

ÇANKAYA UNIVERSITY

BY

SAAD MAZIN SHIHAB AL-DAKHEEL

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF COMPUTER ENGINEERING

INFORMATION TECHNOLOGY PROGRAM

SEPTEMBER, 2019

Title of the Thesis: **UNMANNED SURFACE VEHICLE WITH ANALYTICAL DROWN MAP SYSTEM**

Submitted by: **Saad Mazin Shihab AL-DAKHEEL**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.

Prof. Dr. Can ÇOĞUN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Assoc. Prof. Dr. H. Hakan MARAŞ
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Dr. Instructor. Sibel T. ÖZYER
Supervisor

Examination Date: 13 / 09 / 2019

Examining Committee Members:

Prof. Dr. Bülent TAVLI	(TOBB ET Univ .)
Dr. Instructor. Selma ÖZAYDIN	(Çankaya Univ.)
Dr. Instructor. Sibel TARIYAN ÖZYER	(Çankaya Univ.)

STATEMENTOF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Saad Mazin Shihab AL-DAKHEEL

Signature :

Date : 13 / 09 / 2019

ABSTRACT

UNMANNED SURFACE VEHICLE WITH ANALYTICAL DROWN MAP SYSTEM

AL-DAKHEEL, Saad Mazin Shihab

M.Sc., Department of Computer Engineering/Information Technology Program

Supervisor: Dr. Instructor Sibel TARIYAN ÖZYER

September 2019, 51 pages

Drowning occurring both at sea and in the ocean - especially the increasing cases among refugees requires the use of advanced technology and tools to address this problem. The Unmanned Surface Vehicle is a smart boat that can be self or remotely controlled, this boat is hired in marine surveys, explorations, analysis … etc. In this thesis, the analytical drowning mapping system is designed for a hypothetical unmanned surface vehicle with touch sensors installed on the platform and GNSS. This marine vehicle works on detecting the people who are exposed to drowning with their approximate number and location besides the marine vehicle's location. The designed system is provided with an algorithm that works on determining the dangerous zones on the map based on the number of people who hold the boat and their locations. By using the grid property on the map, the dangerous zones algorithm works on determining the dangerous zones into level on the map and colored these zones according to the dangerous level. These areas are colored starts with green until the red color. Where all data is saved for every operation. Due to limited storage and processing capacity of physical units with a large number of data, cloud computing has been adopted to solve this problem. All data and information

generated from the sensor and mapping system will be stored in the cloud in real time. In order to increase the chance of saving the people lives who are at risk of drowning in sea, this thesis has been partially addressed this challenge which have been neglected in previous studies.

Keywords: Unmanned Surface Vehicle, Cloud Computing, Global Navigation Satellite System, Touch Sensor, Big data, Risk Value.

ÖZ

ANALİTİK ÇEKİLMİŞ MAP SİSTEMİ İLE İNSANLI YÜZEY ARAÇI

AL-DAKHEEL, Saad Mazin Shihab

Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı / Bilgi Teknolojileri Programı

Tez Yöneticisi: Dr. Öğr. Üyesi Sibel TARIYAN ÖZYER

Eylül 2019, 51 sayfa

Hem denizde hem de denizde oluşan boğulma - özellikle mülteciler arasında artan vakalar, bu sorunu çözmek için ileri teknoloji ve araçların kullanılmasını gerektirmektedir. İnsansız Yüzey Aracı, kendi kendine veya uzaktan kontrol edilebilen akıllı bir teknedir, bu tekne denizcilik araştırmaları, keşifler, analizler vb. için işe alınır. platformda ve GNSS'de yüklü. Bu deniz aracı, deniz araçlarının bulunduğu yerin yanı sıra yaklaşık sayısı ve konumlarıyla boğulmaya maruz kalan kişileri tespit etmeye çalışmaktadır. Tasarlanan sistem, botu tutan insanların sayısına ve bulundukları yere göre haritadaki tehlikeli bölgeleri belirlemeye çalışan bir algoritma ile sağlanmıştır. Haritadaki grid özelliğini kullanarak, tehlikeli bölge algoritması, tehlikeli bölgeleri haritada düzeye belirlemeye çalışır ve bu bölgeleri tehlikeli seviyeye göre renklendirir. Bu alanlar renkli, kırmızı renge kadar yeşil ile başlar. Her işlem için tüm verilerin kaydedildiği yer. Çok sayıda veri içeren fiziksel birimlerin sınırlı depolama ve işleme kapasitesi nedeniyle, bu sorunu çözmek için bulut bilişim benimsendi. Sensör ve haritalama sisteminden üretilen tüm veriler ve bilgiler gerçek zamanlı olarak bulutta depolanacaktır. Denizde boğulma riski taşıyan insanların hayatlarını kurtarma şansını arttırmak için, bu tez de daha önce yapılan çalışmalarda ihmal edilmiş olan bu zorluğa kısmen değinilmiştir.

Anahtar Kelimeler: İnsansız Yüzey Aracı, Bulut Bilişim, Global Navigasyon Uydu Sistemi, Dokunmatik Sensör, Büyük Veri, Risk Değeri.

ACKNOWLEDGMENTS

At first, I thank god for everything done in my life and for giving me this chance to complete my study. I would like to specially thank my supervisor Dr. Instructor. Sibel TARIYAN ÖZYER for her helpful advices, valuable comments and guidance to make this thesis completed. Her door was always open for me whenever I need her help. It is a pleasure to express my special thanks to my wings (my parents) and my family for their unlimited supporting all the time. This work is propped by the Turkish grants Kosgeb through the name of the project “Fully Automated Remote-Control Life-Saving Robot”, project code 2F8UV, as a Research and Development Project as well as Cankaya University Technology Transfer Office project No. 1 / KSGB / ARGEI / 2018/04 / 132-1. Special thanks to Rakun company for making this project possible and for their supporting for providing laboratory and project components.

At the end, I would like to thank all the staff in my department at Cankaya University Who support me throughout academic courses.

TABLE OF CONTENTS

STATEMENT OF NON-PLAGIARISM PAGE i
ABSTRACT iv
ÖZ vi
ACKNOWLEDGMENTS vii
TABLE OF CONTENTS viii
LIST OF FIGURES x
LIST OF TABLES xiii
LIST OF ABBREVIATIONS xiiii
CHAPTER 1 1
Introduction 1
1.1. Introduction 1
1.2. Motivation 3
1.3. Aim of Study 4
1.4. Problem Statements 5
1.5. Literature Reviews 5
1.4. Thesis Outline 7
CHAPTER 2 9
Current Based Techniques and Components 9
2.1. Background of USV 9
2.1.2. USV Development 10
2.2. Cloud Computing 12
2.2.1. Service Models of Cloud Computing 13
2.2.2. Cloud Types 14
2.3. Brief of GNSS Work 16
2.4. GNSS Segment 17
2.4.1. Space Segment 17

2.4.2 Control Segment 18

2.4.3 User Segment 18
2.5. Static Positioning 19
2.6. Real-Time Kinematic 20
2.7. LattePanda Alpha 21
2.8. Arduino Mega 24
2.9. The Touch Sensor 25
CHAPTER 3 26
The Thesis Methodology 26
3.1. Introduction 26
3.1.1. Map Creation Process 30
3.1.2. Grid Map System 32
3.2. Hardware Side 34
3.2.1. The Touch Sensor Connection 34
3.2.2. The GNSS Modul Connection 35
3.3. Software Side 36
3.3.1 The USV-Cloud Mechanism 37
3.3.2 The Cloud-Drowning Map Mechanism 38
CHAPTER 4 41
The Experimental Results 41
4.1. Results 41
CHAPTER 5 44
Conclusion and Future Work 44
5.1. Conclusion 44
5.2. Future Work 45
REFERENCES 46

LIST OF FIGURES

Figure 1.1: The total arrivals of people Mediterranean Sea and land who passed to Europe from January 2019 until 29 Jul 2019 [12] 4
Figure 2.1: Fields utilization of Cloud Computing [38]. 12
Figure 2.2: Service models of Cloud Computing [43]... 14
Figure 2.3: Types of Cloud Computing 15
Figure 2.4: GNSS signal trilateration [47] 16
Figure 2.5: Stationary receivers [48].. 19
Figure 2.6: Real-Time Kinematic [50].. 20
Figure 2.7: LattePanda Alpha [51].. 21
Figure 2.8: The Arduino Mega 2560 [54]... 24
Figure 2.9: Capacitive touch sensor [56] 25
Figure 3.1: Internal and external communication layers of the sensors connections [57] 27
Figure 3.2: The general operation’s mechanism [57] 29
Figure 3.3: The generated landmarks on the map [57] 31
Figure 3.4: Illustration of the touch sensor connection [58] 34
Figure 3.5: Arduino code of the touch sensor initialization [58] 35
Figure 3.6: The GNSS connection 36
Figure 3.7: The USV - Cloud mechanism... 37
Figure 38.: The Cloud- Drowning Map mechanism 38
Figure 3.9: The entire USV- Cloud- Drowning Map mechanism............................. 40
Figure 4.1: The landmarks are determined based on the touch sensors and GNSS.. 41
Figure 4.2: The dangerous zone level based on number of landmarks per pixel...... 42
Figure 4.3: The dangerous zones 43

LIST OF TABLES

Table 2.1: The specifications of the LattePanda [51]..23
Table 3.1: Grid variable..33

LIST OF ABBREVIATIONS

USV	: Unmanned Surface Vehicle
GPS	: Global positioning System
GNSS	: Global Navigation Satellite System
GPS	: Global Positioning System
Paas	: Platform as a service
Iaas	: Infrastructure as a service
Saas	: Software as a service
RTK	: Real-Time Kinematic

CHAPTER 1

INTRODUCTION

1.1 Introduction

The Drowning Map System is a surveillance system used to identify people who are prone to drowning in the sea by using the unmanned surface vehicle, sensors and other devices. The USV (unmanned surface vehicle) is a vessel that is remotely controlled by an operator located in another place such as a vehicle or other land. USV that has made a quantum leap in the maritime field is used for various purposes like survey, research, discovering and more, USV consists of physical parts (like actuators, sensors and other components). The USV can play different roles depending on how they are hired it based on the need for use. Accordingly, USV will be employed to undertake exploration tours to discover and try to rescue people who are prone to drowning in the sea, especially after increased drowning cases among refugees in order to provide them with rescue teams in a timely manner. This can be done by adding some sensors to the vehicle to implement this operation.

The renowned GPS (Global Positioning System) is a U.S. owned auxiliary tool that works on provide information to the users on position, navigation, and timing (PNT) for a various set of applications [1]. The GPS solved many of the problems facing the vehicle represented in improving the vehicle's control system by providing precise geographic information for the vehicle. The Global Positioning System (GPS) is one component of the Global Navigation Satellite System (GNSS). GNSS is a comprehensive term covering all global satellite positioning systems. This includes groups of satellites orbiting the Earth's surface and continuously transmitting signals that enable users to locate them. In addition to GPS, there are other satellite navigation systems, such as the Russian

GLONASS system, and other systems that may soon include such as Galileo in the European Union and Beidou in China. All these systems are all under GNSS system [2].

Although the GNSS and GPS are work together there is a major difference between GPS and GNSS lies in that navigational satellites can be used by GNSS-compatible equipment from other networks outside the GPS system. That's means more satellites leads to increase receiver accuracy and reliability [3]. So, if the GPS satellites suddenly stop working and you are using a GNSS receiver, it would switch over to GLONASS satellites without feeling of the user to the difference [4].

In addition to setting the target point and the route of the tour of the USV, it's also locating the current vehicle during the rounds. The collision avoidance system (CAS) can also get benefits from GNSS by locating obstacles on the map after being detected by radar, LIDAR and ultrasonic sensors [5].

GNSS can be used to locate potential sinking victims on the map with the help of the touch sensors. The touch sensor is a sensor that measures the applied force on itself. The touch sensor can be a tape or a circle shape and therefore it is efficient, effective, inexpensive and easy to handle.

The installation of touch sensors onboard the vessel can be useful for detecting the persons who are compressing them while holding the boat based on the applied force on the sensors and locating them by GNSS.

All the sensors and actuators that have been mentioned earlier and more make up networks of interconnected objects called IoT (Internet of things), in order to get information about the world around us, and these things or things exchange and communicate with one another, however, the capacity of these objects for storage and processing of are limited [6].

The sensors that have been mentioned above in addition to the USV control system, collision avoidance system and more sensors must be integrated into one system which is the main computer. The main computer is a system that responsible for organize the way of sensors work and transit the information of these sensors to the surveillance system in a real-time. It collects information on the vehicle state (location, position, routing, climate…etc.), also, touch sensors status when its active or not. All this information is updating continuously from the main computer to the surveillance system.

The sensors and the real-time surveillance system are generating data continuously at the same time. That extreme number of data requires a massive storage and process capacity, the physical storage units are expensive, exposed to damage and need to be updated and maintained continuously. Therefore, a cloud computing is adopted to solve these issues. Cloud Computing it's provides a pool of services such as host, control, unlimited storage and processing capabilities [6][7], so that we can upload the generated data and information to the cloud in real-time. Since the cloud is providing flexibility in host, control and databases for data and application, the drown map system (surveillance system) can be also hosted on the cloud.

The massive amount of data that are generated are scientifically known a Big data. These data are various in format due to data sources. The database system instruments of Big data that are used for storing, capturing, analyzing and processing data are much better than of the regular one [8] [9]. Accordingly, Big data analysis helps improve decision-making and understand patterns of different data types after converting data into a format that can be understood using tools and techniques [10]. Therefore, the drown map system can use the big data techniques to analyzes search operations (routes, locations, landmarks, number of people …etc.) and makes hazardous zones on the map. According to the number of people who holdings the touch sensor that installed on the USV after identified their location based on GPS sensor, hazardous zones on the map are colored into three areas start with Green to the gradients of red.

1.2 Motivation

According to united nations high commissioner for refugees (UNHCR) [11] a 21,591 have risked their lives reaching Europe by Mediterranean Sea so far 7th Jun of 2019 that includes refugees who seeking for a safe shield a way of conflict zones and migrants arriving by sea to Italy, Greece, Spain, Cyprus and Malta. A 530 were estimated dead and missing (in sea and land), and it was more in the four years that preceded it [12]. These alarming numbers of statistic in addition to paucity of the previous studies of employing the USVs in rescue sector, as well there are no enough reliance on advanced technologies like cloud computing, Big data and maps in the rescue sector.

These advanced technologies can be used to identify and analyzing the locations where drowning cases are occurred based on current and previous information in order to reduce these cases. All these reasons led to the motivation of creation of this thesis.

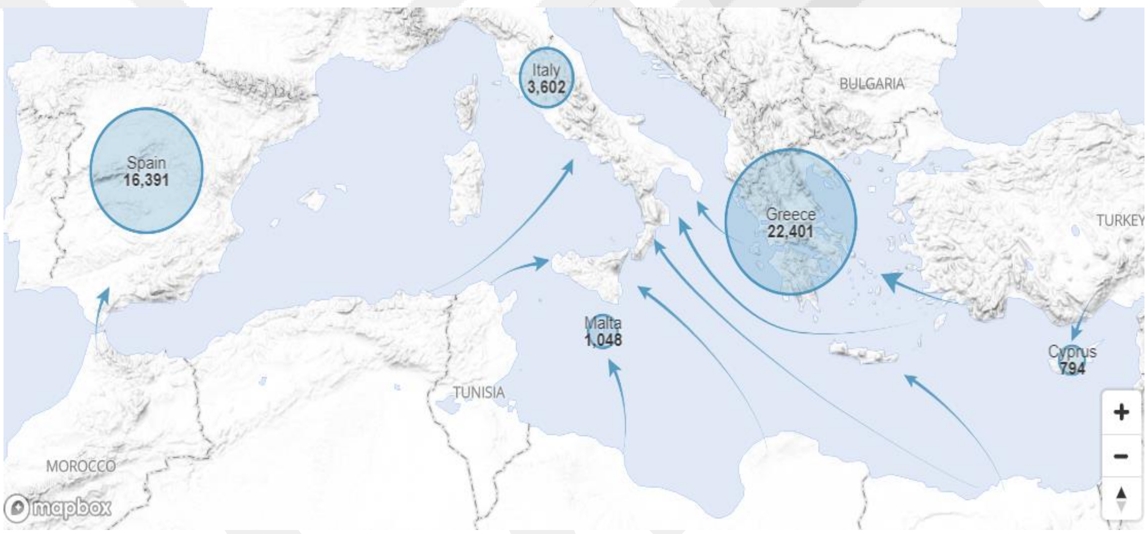

Figure 1.1: The total arrivals of people Mediterranean Sea and land who passed to Europe from January 2019 until 29 Jul 2019 [12].

1.3 Aim of Study

The thesis is aiming to save people's life who are prone to drown in the sea whether refugees who trying to cross the sea to the other side or for people who spending their time right there for various purposes like fishing. Also, the thesis contributes in improvement of the watercraft operations based on the sensors, radars and main computer that equipped with and the surveillance system. It is also works on presentation of Big data and cloud computing properties by applied it with surveillance system. The map drown system contributes in display and clarification the information of the conducting operations, as well, it contributes in determining the dangerous areas on the map after analyzing the collected data based on the big data and cloud computing techniques.

1.4 Problem Statements

The experimental side may face some barriers in order to be implemented, these barriers lies in: first, integrating the sensors and radars to works based on each other under on system. Second, determining the approximate number of people who hold the boat by determine the average value of human grip based on the applied force on the touch sensor. Third, determining the location of the people when they hold the boat. Four, make the dangerous zones on the map by analyzing data according to momentum zones. Five, presenting the information of the previous operations of the USV (tours, location, dangerous areas …etc.) when required.

1.5 Literature Reviews

In paper [13] the authors presented a USV model called Emily that is used in lifesaving of drowning victims in the open water. Authors adopted an approach that implementing an artificial potential field with constant, linear, and exponential magnitude profiles to navigate EMILY autonomously to a goal point based on the GPS. This approach aims to decelerate the USV when the USV is approaching the victim's site. The artificial potential field consists of vectors that representing the action of the USV motor. Where each vector composed of a theta and a magnitude component representing the desired movement of the robot. This vector is calculated by comparing the GPS of the USV with the GPS goal to be translated to commands in order to control the USV.

In [14] a wireless Unmanned Surface Vehicle (USV) is designed for monitoring water-quality to solve the river pollution issue. A communication USV sub-system was designed to long-distance broadcast purpose by using HTTP and TCP communication protocols depend on 4G communication technology. On the control side, data interface and TTL interface were used in the USV to achieve data transition between control and sensors. The USV was equipped with main hardware that contains PH sensor, TDS sensor for water quality data, GPS positioning unit, 9-axis sensor, ultrasonic for automatic cruise and obstacle avoidance. WIFI, Bluetooth and 4G communication unit were adopted for communications.

A 32- bit MCU and Arduino control board based on ARM Cortex-M3 core to integrate and build the system to get water-quality data and automatic cruise. A Mobile application software was built to monitor and display the real-time position and accurate moving paths of USV and get the collected data of water- quality. Users can send location data to the USV, the USV begins to move on the predefined route on app of smart terminal and collect water quality data once the USV arrives at the target location and send data to the server.

In [15] a solution was proposed for the problem of GPS Abnormality. The authors proposed an approach that is fusing multiple sensor measurements of Unmanned Surface Vehicle (USV) to estimate the location of the USV and discover the GPS interference in the same time. This integration led to both better detection and navigation of GPS interference. To find the attitude and location of the USV, the method fuses available sensor measurements by GPS, magnetic field, angular rate, acceleration, linear velocity, and range and bearing to acoustic beacons through Extended Kalman Filter (EKF).

For detection of GPS outlier and interference, this method uses error variation of estimated location. Where GPS interference is find out by comparing the location estimated by fusing multiple sensor measurements with the location detected by GPS. The method excludes GPS data from navigation process when outlier and interference of the GPS is detected. This method was tested using measurement and simulated data that produced by navigation of a ground vehicle. The results showed that the method detects GPS recovery as well as the GPS interference and outlier, which liberates USV navigation from GPS faulty problem.

In [16] the authors presented how the data partitioned into cells to give priority to Geo-spatial location. The area geography like a country, state, district or all kind of area in world is divided into data cells. Depend on the location specific queries on the area that has been previously observed, the data cells size is decided. The size of the cell is determined so that the queries majority are addressed within the cell itself. This allows computation to occur closer to the location of data. As a consequence, the elimination of the data communication overheads is achieved.

The authors have also created some data redundancy, that used not just to allow failure mechanisms happen but also to aim the performance. This operation is done by a methodology called 'nine-cell' that each cell stores data of its own along with eight of its neighbor's data. The overloading of queries in cells can be easily solved by passing off some of their workload to the neighbors that are near of them and guarantee timing in response. Additionally, load balancing of data effectively guarantees better resources utilization. The results of experimental side proved that the adopted approach has improved query response times, introduced better productivity and away from enabling updates in real-time on data, also the adopted approach has reduced the average of query waiting time.

1.6 Thesis Outline

- In the chapter two, the current based techniques and components were introduced. The background and development history of USV, Cloud Computing with its services and models, GNSS module including GNSS segments and types (static and real time kinematic) and the method of its work, Touch sensors, Lattepanda Alpha and Arduino Mega were presented and discussed.

- In the chapter three, the Methodology of work was introduced. The map creation and grid map system were explained. The hardware components and the method of their connection were presented. The methodology of software design and work was presented as well.

- In the chapter four, the results were illustrated.
- In the chapter five, the conclusion and the future work possibilities of this study were discussed.

CHAPTER 2

CURRENT BASED TECHNIQUES AND COMPONENTS

2.1 Background of USV

The first development of the USVs was during World War II for martial operations [17]. In the next two decades, the targets of firearms and missiles were the most important uses of USVs [17]. In addition to military use, the USV's educational and civic deployment began to arise in the 1970s [18]. the USVs usage has been widened since the 1990s with a number of significant research and development programs. The USVs have made an important role in recent years being published for actions as varied as coastal line patrol, ocean exploration and environment monitoring [19].

Today's USVs have been improved and showed by academic laboratories, government users and companies. Tasks provided include bathymetric mapping, defense, general robotics research and science [20]. By mid-2016 comparatively classical vessels designed to run without staff are providing by industrialists. Either equipped with some meaningful autonomous capabilities or remotely controlled these systems are supporting numerous task demands [21].

The NGC system is very important, to accomplish the task by guiding and controlling USV motion.

The general NGC system contains three subsystems: navigation, guidance and control systems. Based on different marine navigation sensors to determine the USV's location, speed and course, the navigation subsystem watch and gains real-time traffic information. The responsible guidance subsystem presented in generating safe trajectories, in accordance with the particular task using information provided by the navigation sensor subsystem. The propulsion system is managed by the control subsystem, in a USV this is the rudder angle and propeller thrust (usually by motor speed) while preserving vehicle fastness. The alternate order may be differential orientation [19].

with the assistance of more built-in, effective, commercially affordable and available navigation tools, including inertial measurement units (IMUs) and global positioning systems (GPSs), also more reliable and powerful wireless communication systems [20], major opportunities were provided for USVs and their applications than past time. for a spacious range of possible applications in a cost-effective way, like environmental missions, scientific research, military uses, ocean resource exploration and other applications, USVs can be developed [22].

2.1.2 USV Development

In the mid-1990s command line interfaces were adopted to programming the unmanned vehicles and layered 6s

control approaches were used. The graphical user interfaces in these days are typical. But more deeply thing is the development of powerful "self-control" algorithms and automation "operating systems" available out of open source channels. One of the offering that has significantly increased the ability of novel UMV developers to field reliable offerings quickly is the Mission Oriented Operating System (MOOS).

Progressed software systems have expanded the range of operators and functions available. The SeeByte autonomous motor Neptune can conduct real-time monitoring of many unmanned vehicles charged to attain a common goal.

One operator to numerous UMVs is a strong evolution enabled by software [21].

Based on functional applications, USVs may come in a diversity of functionalities and appearances. However, the basic elements that must be included in every USV are following below:

Data collection equipment, GNC systems, Hull and auxiliary structural elements, Communication systems, Propulsion and power system and Ground station [22].

The leading research institutions and universities in the Europe and USA have led the applications non-military of USVs. At Massachusetts Institute of Technology (MIT) located in the USA, the family of USVs was developed is consisting of vehicle 'ARTEMIS' which like a fishing trawler, AutoCat [23] the catamaran Autonomous Coastal Exploration System ('ACES') [24], the kayak 'SCOUT' (Surface Craft for Oceanographic and Undersea Testing) [25].

These USVs demonstrated the feasibility of the automatic heading control using Differential Global Positioning System (DGPS) depended on waypoint navigation and Proportional-Derivative (PD) actions and the possibility of running marine vehicles independently to collect hydrographic data.

"Delfim" the autonomous catamaran was developed originally by the Lisbon DSOR lab (IST-ISR) as a follow-up to the AUV companion in the ASIMOV project funded by the European Union [26]. Then, as a stand-alone unit, it was used to collect marine data and bathymetric maps.

At ISEP (Institute of Engineering of Porto) the Autonomous Systems Laboratory proposed 'ROAZ' the autonomous catamaran with the design objective of supporting missions of AUV [27]; [28] to strongly command the USV.

The University of Plymouth in UK has developed the autonomous catamaran 'Springer' [29]. 'Springer' was designed primarily for undertaking hydrographical and environmental surveys and pollutant tracking in shallow water [30]. Other academic and scientific institutions are also used 'Springer' as a basic test for validation of the newly control algorithms that designed by them [19].

Kaizu et al. [31] have developed an unmanned air-boat for mapping shallow water quality with a maximum speed of 1.2 m/s.

'RB-26' a radio controlled agricultural USV was developed The Hokuto Yamar, a Japanese company, to move in the rice field for the purpose of autonomous fertilization and weeding [32].

CNR-ISSIA Genova (Italy) has developed ‘Charlie’, the autonomous catamaran to collect the biological elements of the sea surface in the micro-layer surface [33].

Li and Weeks [34] have used a USV to measure currents at the tidal entrance by equip the USV with an acoustic current profiler.

Twichell et al. [35] have developed a commercial USV to create a map with three-dimensional feature of an oyster habitat along the shallow coast.

2.2 Cloud Computing

The disruptive advanced technology "Cloud Computing" has solutions for the big number of sensors and the generated data of these sensors by provides unlimited storage and

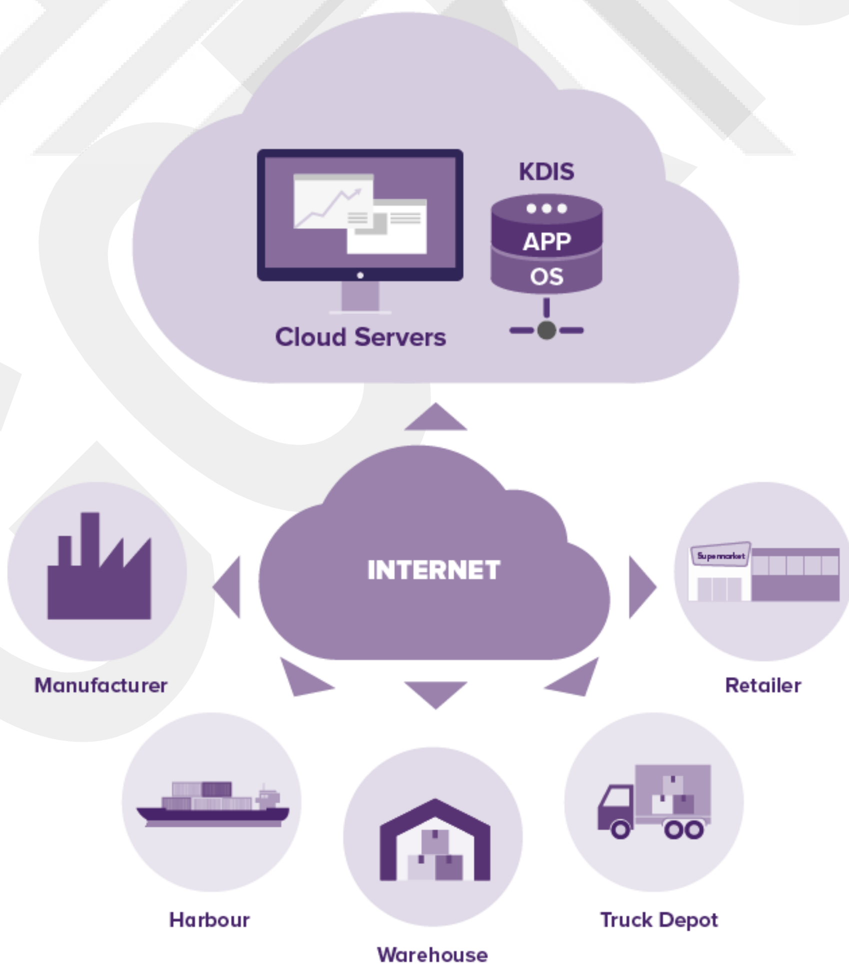

Figure 2.1: Fields utilization of Cloud Computing [38].

computations capabilities [6] [36]. Cloud computing was defined by the National Institute of Standard and Technologies (NIST) [37] as ‘‘Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction’’. Cloud computing resources can be rented in on-demand mode where these resources are being provided as general services [7]. Large companies (such as Alibaba, Google, IBM, etc.) have widely relied on this model to deliver online services and gain economic and technical advantages.

Still there are some problems that face cloud that presented in security, privacy and service-level agreements [7] [39]. Due to features and low-cost, Alibaba cloud is adopted in this thesis.

2.2.1 Service Models of Cloud Computing

Generally, cloud services are grouped into three main types from above to below, each of these services provides special services and functions. These types are: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

Four layers are form the architecture of Cloud Computing: infrastructure, platform, application (software) and datacenter (hardware) [40].

There are three models of services of Cloud as it shown in Figure 3, which are [41]:

Lowest layer \(Iaas) infrastructure as a service: provides accessing, processing, monitoring and managing remote data center infrastructures like networking, storage, computing and network services [7] [42].

Middle layer \(Paas) platform as a service: deploying applications, operating system support, software development frameworks, etc. PaaS develops, test, and deploy applications in a fast, simple and cost-effective manner. Using this technology, enterprise operations, or provider of a third-party, are able to manage operating systems, servers, networking, storage, virtualization, and PaaS itself [42].

Highest layer \(Saas) software as a service: provisioning of deliver applications by using web where these applications running on Cloud environments (user applications). Without the need for downloads or installations, most of these applications can be run straight from a web browser [7] [42].

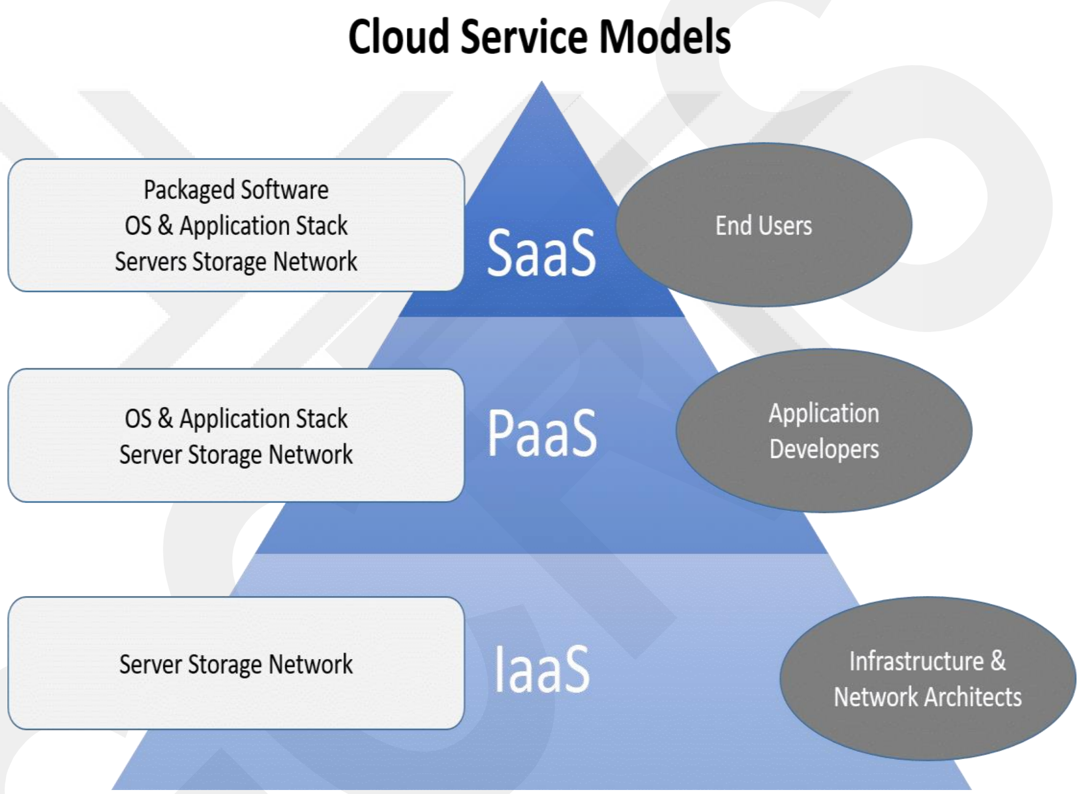

Figure 2.2: Service models of Cloud Computing [43].

2.2.2 Cloud Types

According to [37] [40] there are five popular types of cloud (Figure 2.3):

(i) Private Cloud: which mean the resources and applications are located and used within a single organization with dedicated access.

(ii) Community Cloud: can either be a conglomeration of several providers ("provider community cloud") or a cloud infrastructure used communally by a particular user group (for example branch offices or associations) known as a "customer community cloud".

(iii) Public Cloud: cloud computing sources are publicly available and accessible; resource pools are located in the custody of the cloud supplier.

(iv) Hybrid Cloud: it's combination of private cloud and public cloud architectures as needed.

(v) Virtual Private Cloud: this cloud allows business owners to control and configure network settings as required (e.g. security, topology… etc.).

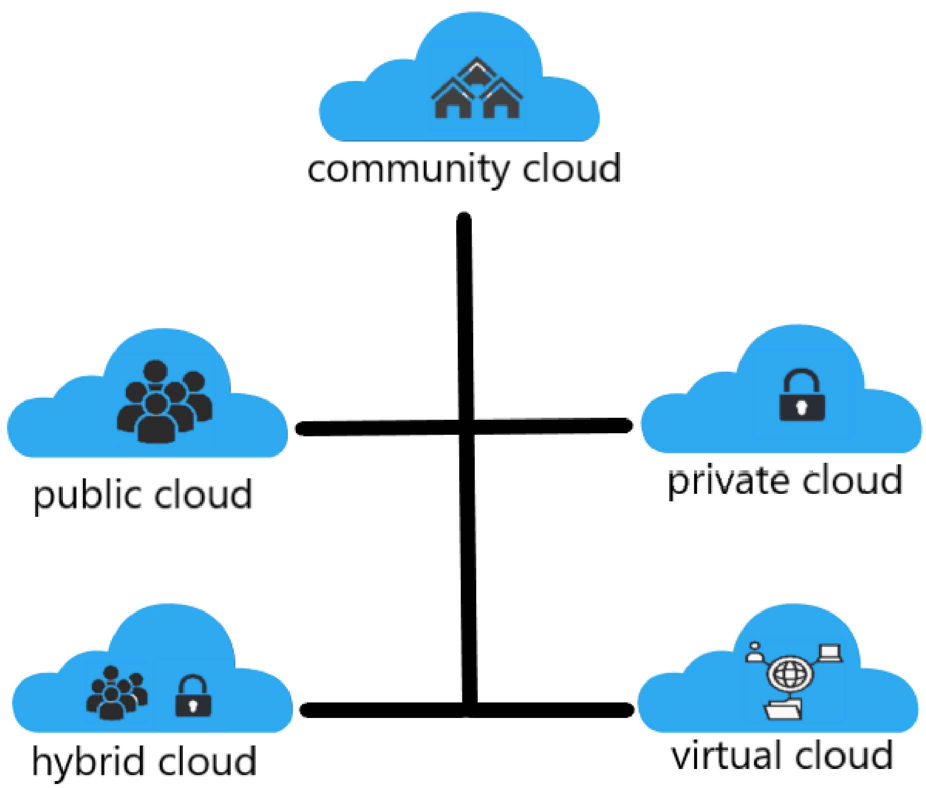

Figure 2.3: Types of Cloud Computing.

2.3 Brief of GNSS Work

The Global Navigation Satellite System (GNSS) methodology encompasses a constellation of all global satellites positioning systems (such as GPS and other satellites) orbiting Earth. These satellites send signals continuously that make the users able to determine their three-dimensional (3D) position and provide global coverage [44]. The GNSS presently involves several satellite navigation systems such as the American GPS, BeiDou (china), the Russian GLONASS and Galileo (European Union) [45].

Based on the basic principle of resection, the Satellite-based positioning works by using either range differences or ranges from a user to a number of satellites in order to calculate a position [46]. The position of the satellites in a reference frame is known. The calculation of the transmission time from the satellite to the user, and thereafter the range is done by the coded signal that transmitted by the satellites toward the user. These ranges and user location can be used by trilogy, as shown in Figure 2.4. Each range of satellite receivers defines the surface area of potential locations, each pair of spheres identifies a circle of

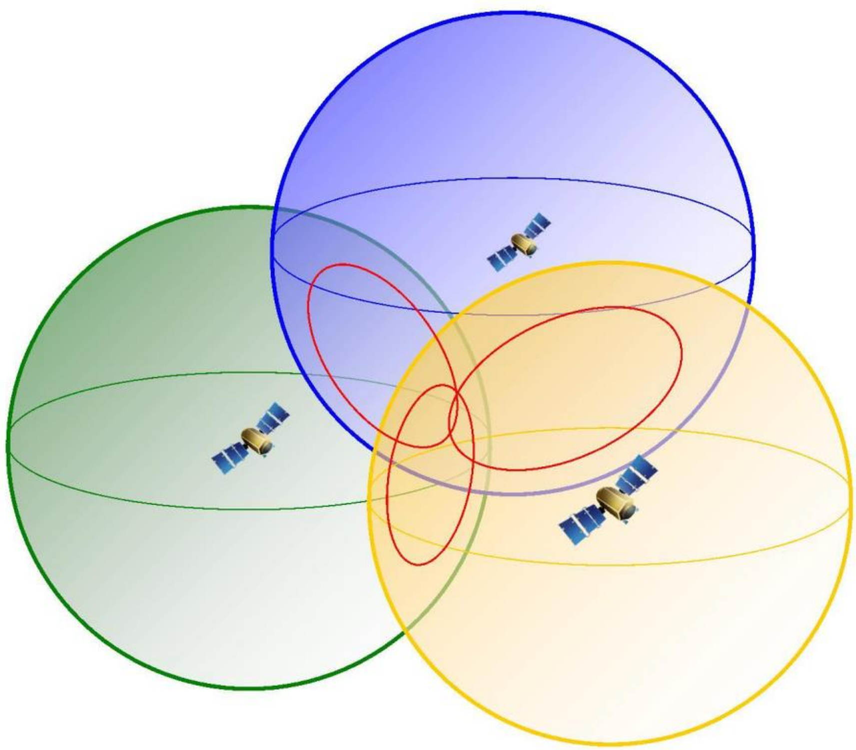

Figure 2.4: GNSS signal trilateration [47].

potential positions between them, and the intersection of these three circuits determines the location of the receiver. Since the user's receiver does not usually have a clock that can provide accurate system time, a fourth satellite is needed to calculate this unknown extra time [47].

The basic that can be observed in the GNSS system is the time required for a signal to travel from the satellite (transmitter) to the receiver. The multiplying between the speed of light (c) by this travel time provides a measure of the distance or apparent range known as pseudorange (R) between them [44].

$$distance = time \times speed$$

The objective is to determine the clock offset (δt) and the receiver coordinates r = (x, y, z) from pseudorange measurements (R) of minimal four satellites in view. The principle of positioning depends on solving a geometric issue from satellites to measured ranges, with known coordinates. Based on the broadcast message, the coordinates of the satellite can be calculated, which by turn provides all the information necessary to model the measurements for the Standard Positioning Service [44].

2.4 GNSS Segment

The GNSS Segment consist of the following segments below:

2.4.1 Space Segment

Generating and transmitting code and carrier phase signals are the main functions of the space segment, also storing and broadcasting the navigation message that was uploaded by the control segment. The control of these transmissions is done by the very stable atomic clocks onboard the satellites.

The satellite constellations with enough satellites form the space segments of GNSS to guarantee that users will have at least four satellites at the same time from any point on Earth's surface at any time [44].

2.4.2 Control Segment

The control segment (also known as the ground segment) controls the correct operation of the GNSS system. Its core functions are [44]:

- To maintain and control the configuration and status of the satellite constellation.
- To predict the astronomical calendar and evolution of the satellite clock.
- To maintain the corresponding GNSS time range (through atomic clocks).
- Updating navigation messages for all satellites.

2.4.3 User Segment

The user segment consists of GNSS receivers. Its main function is to receive GNSS signals, identify pseudorange (and other observations) and in order to obtain the coordinates and provide a very accurate time, the user segment is also solving the navigation equations.

The main elements of the general GNSS are: an intermediate-precision oscillator, a microprocessor, an antenna with pre-amplification, a radio frequency section, some memory for data storage and an interface with the user, a feed source. The calculated position is referred to the antenna phase center [44].

2.5 Static Positioning

It is positioning technique that rely on the measurements of the carrier-phase. It could be used in two or even more stationary receivers at the same time that monitoring the satellites as shown in the figure below (Figure 2.5). It can be noticed there is one receiver, the base station coordinates should be pre-defined i.e. known point. The base receiver could be known, as long as at least there are four common satellites is visible [48].

The measurement of the algorithm relies on the nature of the receiver's antenna. The static positioning techniques have resulted greater accuracy, as multiple observations could be made at the same location. Processing the system for long period of time, data that collected could be reduced the errors and biases of the position that achieves a greater condense [48].

The state vector to be determined is given by:

$$X = (dx, dy, dz, t)$$

where the coordinates are considered as constants due to the steady of the receiver and the clock offset can be designed as white noise with zero mean [49].

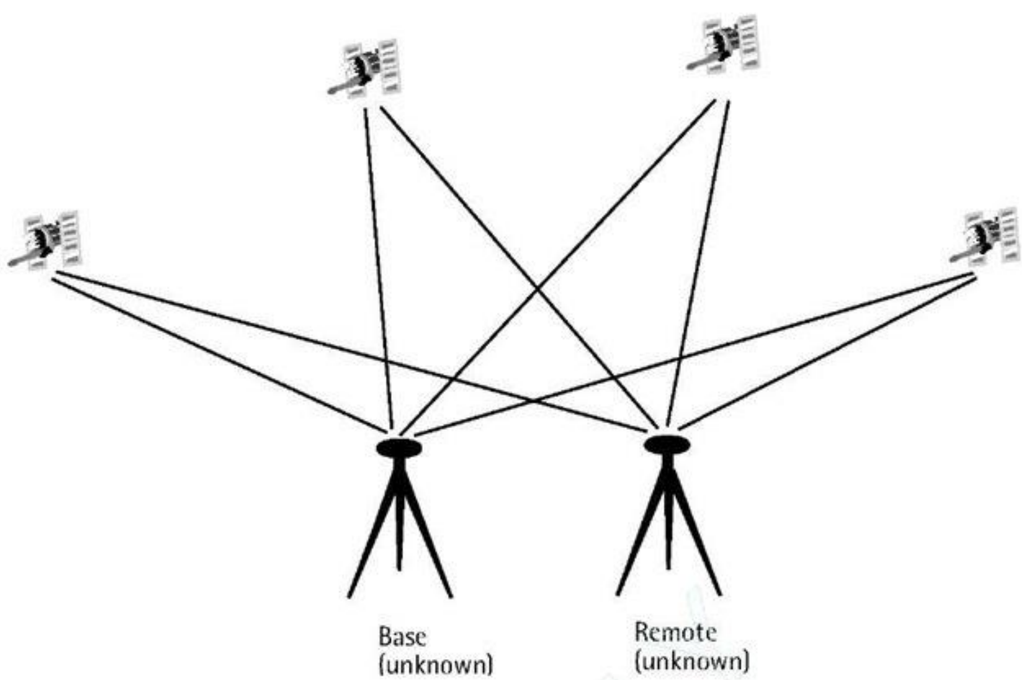

Figure 2.5: Stationary receivers [48].

2.6 Real-Time Kinematic

The Real-Time Kinematic or referred as RTK is a way that provides positional accuracy almost as static carrier phase positioning efficiency but faster. RTK performs real-time positioning, (Figure 2.6). It involves the use of at least one moving receiver (the rover), at least one stationary reference receiver, and the base station. All receivers concerned are observing the same satellites at the same time. The base receivers are constant on control points. The rover moves from the project point to the project point and stops for moments at every new point, commonly a little while. The Vectors are provided by the collected data between the base receivers and themselves as shown in Figure 2.6 in real-time.

When compared the real-time kinematic with the other relative positioning methods, there is no doubt that the very short sessions of the real-time kinematic method are able to produce the largest number of positions in the least amount of time.

The carrier phase ranges do the RTK, where five satellites at least must be tracked by RTK.

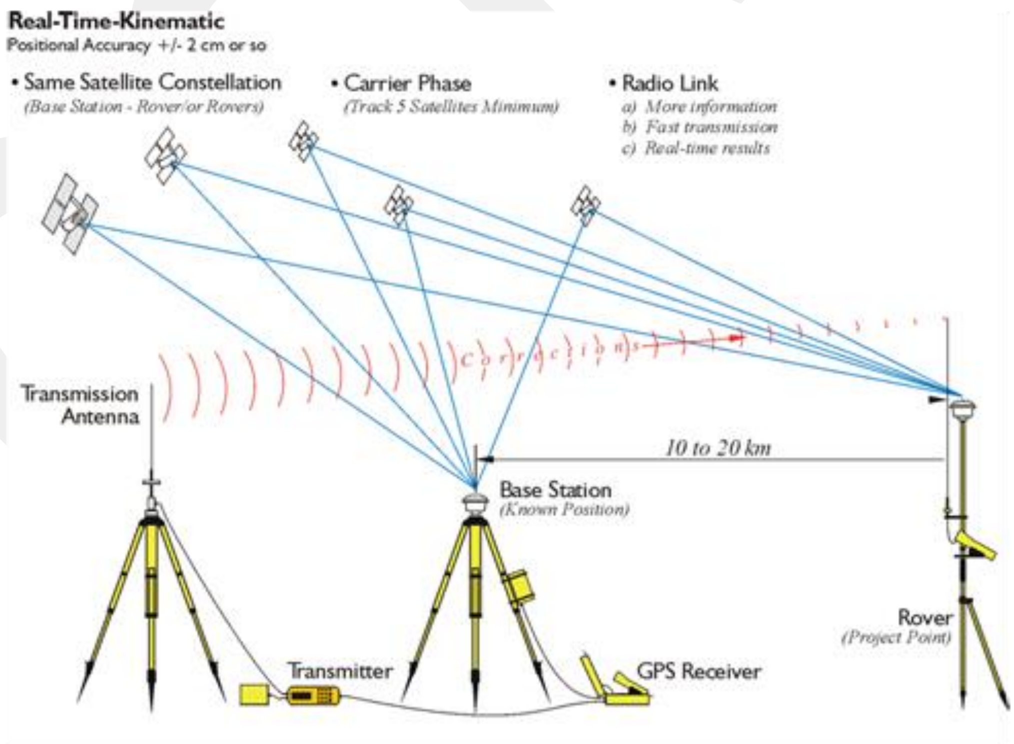

Figure 2.6: Real-Time Kinematic [50].

Tracking five satellites essentially is a reason to make sure of having a position all the time by providing one spare.

On a known point, the base station was built. The base station that associated with the transmission antenna, where a radio transmitter sends corrections to a rover through it [50].

2.7 LattePanda Alpha

The advanced powerful computer LattePanda (Figure 2.7) runs Windows 10 and compatible with Linux, using the same Intel Core m3 processor as a MacBook. It's a second-generation SBC, a fully functional computer, but its small size opens up a whole range of possible uses. LattePanda Alpha aims to collect laptop performance in a small SBC. The LattePanda team has reduced this computing monster from a laptop to a phone size.

The LattePanda Alpha designed from the ground up to run a full version of Windows 10 as standard. Additionally, a Linux compatibility were added due to the different needs of users that was noticed from 1st Generation products. Based on Windows 10 Home, a Windows 10 Pro was produced with enhanced operating system from Microsoft that keeps all the features of the home version, consistent with all Windows software, whereas

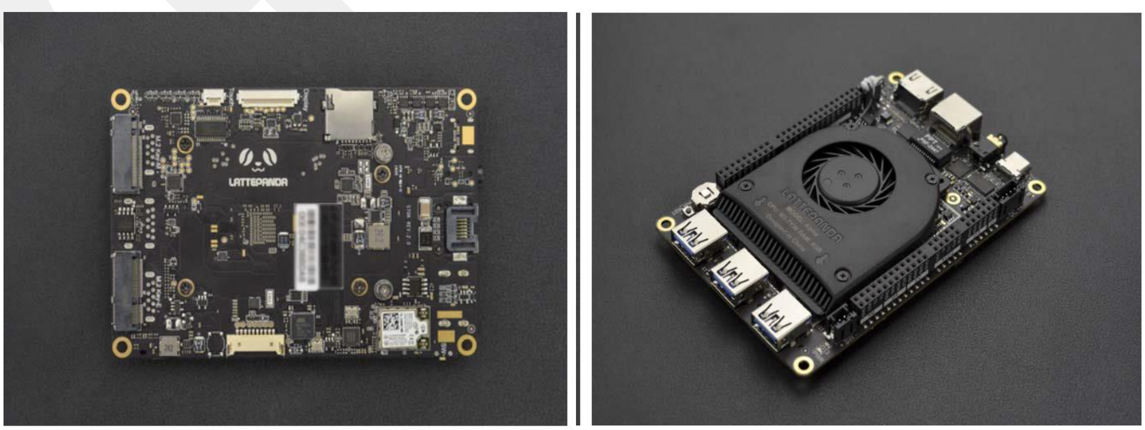

Figure 2.7: LattePanda Alpha [51].

providing greater development applications, remote control and security for industrial applications such as management features and remote desktop.

It makes the LattePanda Alpha product suite compatible with the productivity tool requirements. One of the unique features of LattePanda Alpha, which distinguishes it from competition, is the inclusion of the Arduino Leonardo chip on the board itself. Where it still has all the functions of a traditional Arduino board. This will attract the hobbyist user because it provides straight control and provides simplifying of writing programs to the onboard micro controller. The fact that they are integrated into the same motherboard increases system portability and eliminates the need to load an additional panel. This makes it easy to add high-speed solid-state storage upgrades and the process of doing so is much simpler to upgrade, making Alpha more adaptable appliance to the requirements of the user [51] [52]. The specifications of the LattePanda Alpha can be shown in the table below (Table 2.1) [51]:

Table 2.1: The specifications of the LattePanda [51].

CPU	Intel 8th m3-8100y
Core	1.1-3.4GHz Dual-Core，Four-Thread
Benchmark (PassMark)	Up to 4128
Graphics	Intel HD Graphics 615, 300-900MHz
RAM	8G LPDDR3
Memory	64GB eMMC V5.0l
External Memory	1x M.2 M Key, PCIe 4x, Supports NVMe SSD and SATA SSD 1x M.2 E Key, PCIe 2x, Supports USB2.0, UART, PCM
Connectivity	WIFI 802.11 AC, 2.4G & 5G Dual Band Bluetooth 4.2 Gigabyte Ethernet
USB Ports	3x USB 3.0 Type A 1x USB Type C, supports PD, DP, USB 3.0
Display	HDMI Output Type-C DP Support Extendable eDP touch displays
Co-processor	Arduino Leonardo
GPIO & Other Features	2x 50p GPIOs including I2C, I2S, USB, RS232, UART, RTC， Power Management，Extendable power button
OS Support	Windows 10 Pro Linux Ubuntu
Dimension	115mm * 78mm * 14 mm

2.8 Arduino Mega

The advanced device Arduino Mega 2560 (Figure 2.8) is a microcontroller board depend on the ATmega2560. The Arduino Mega 2560 provided with a 54 digital input/output pins (14 of which allocated as PWM outputs), 4 UARTs (hardware serial ports), 16 analog inputs, a USB connection, a 16 MHz crystal oscillator, an ICSP header, a power jack and a button for reset. The Arduino Mega 2560 has all the necessary components to support the microcontroller. With a feature of easy connection to a computer by a USB cable or

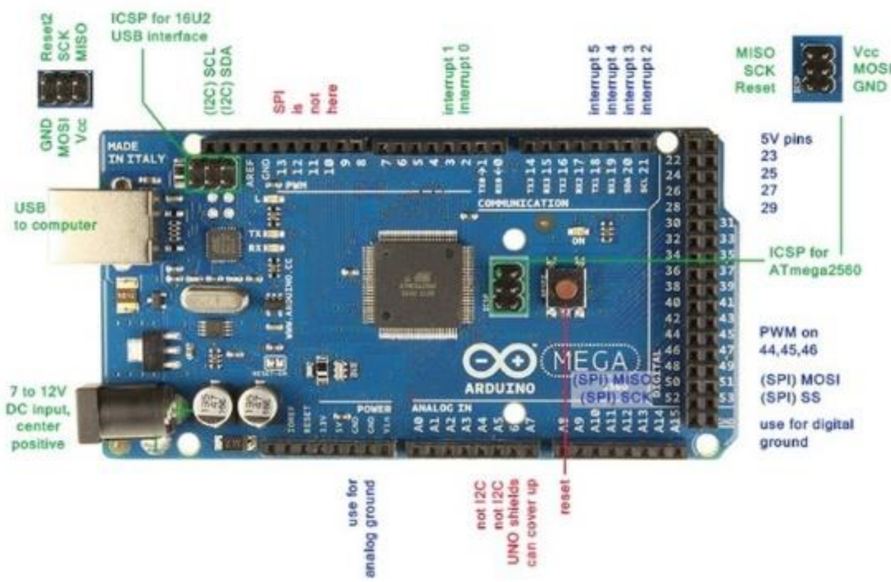

Figure 2.8: The Arduino Mega 2560 [54].

by an AC-to-DC adapter or battery to power it to get started, where the source of the power is selected automatically. The board can operate on an external supply of 6 to 20 volts; the recommended range is 7 to 12 volts.

For the storing code purpose, the ATmega2560 has a flash memory up to 256 KB (the bootloader uses 8 KB of which), 4 KB of EEPROM (that can be written and read with its library) and 8 KB of SRAM. The Arduino Mega 2560 provides a several types of communication with another Arduino, microcontroller or computer [53].

2.9 The Touch Sensor

The TTP223 (Figure 2.9) is an IC touch panel detector that provides a single touch switch. IC Touch Detector is designed to replace the traditional direct button switch with a variety of panel size. The capacitive touch feature lets electronics feel when your finger is a few millimeters away from the surface to simulate a "press" button just like the button press. Capacitive sensing can be used anywhere where the sensation of human touch is low-power or undesirable [55].

The specifications of the touch sensor are below [55]:

- On-board TTP223 capacitive touch a single bond induction IC.
- Size of the PCB board: 29mm x 16mm.
- Working voltage: 2.0 V to 5.5 V.
- Board level indicator.

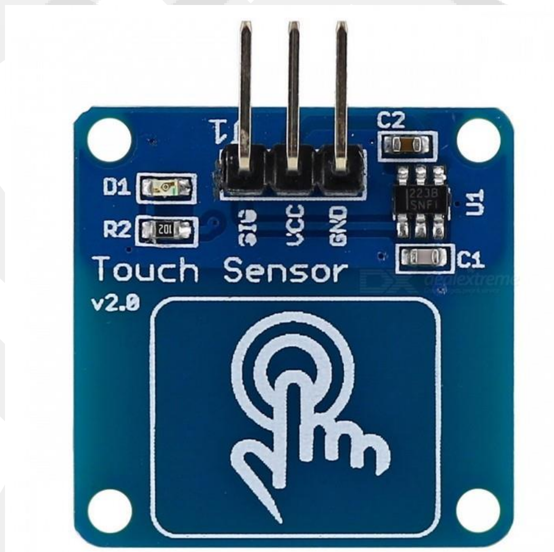

Figure 2.9: Capacitive touch sensor [56].

CHAPTER 3

THE THESIS METHODOLOGY

3.1 Introduction

The thesis methodology is to build a prospect drowning surveillance system of the cases that occurs in sea depends on a USV that equipped with a main computer to transmit the information of monitoring operations to the control center in real-time. Also the USV is equipped with a various kinds of sensors to generate this information. The control center includes information on the unmanned watercraft to be used in purposes of life-saving controlled by an operator. The system is responsible for detecting the person's attachment to the USV at the time of operation when a person is picked up. The system records and processes this data, also produces an interactive drowning map with the analysis of this data.

Based on this drowning map, the drowning hazard data on the coast of unmanned marine vehicles is handled by the system interactively on a map in order to determine the repeated drowning cases in terms of time, areas and actions like this.

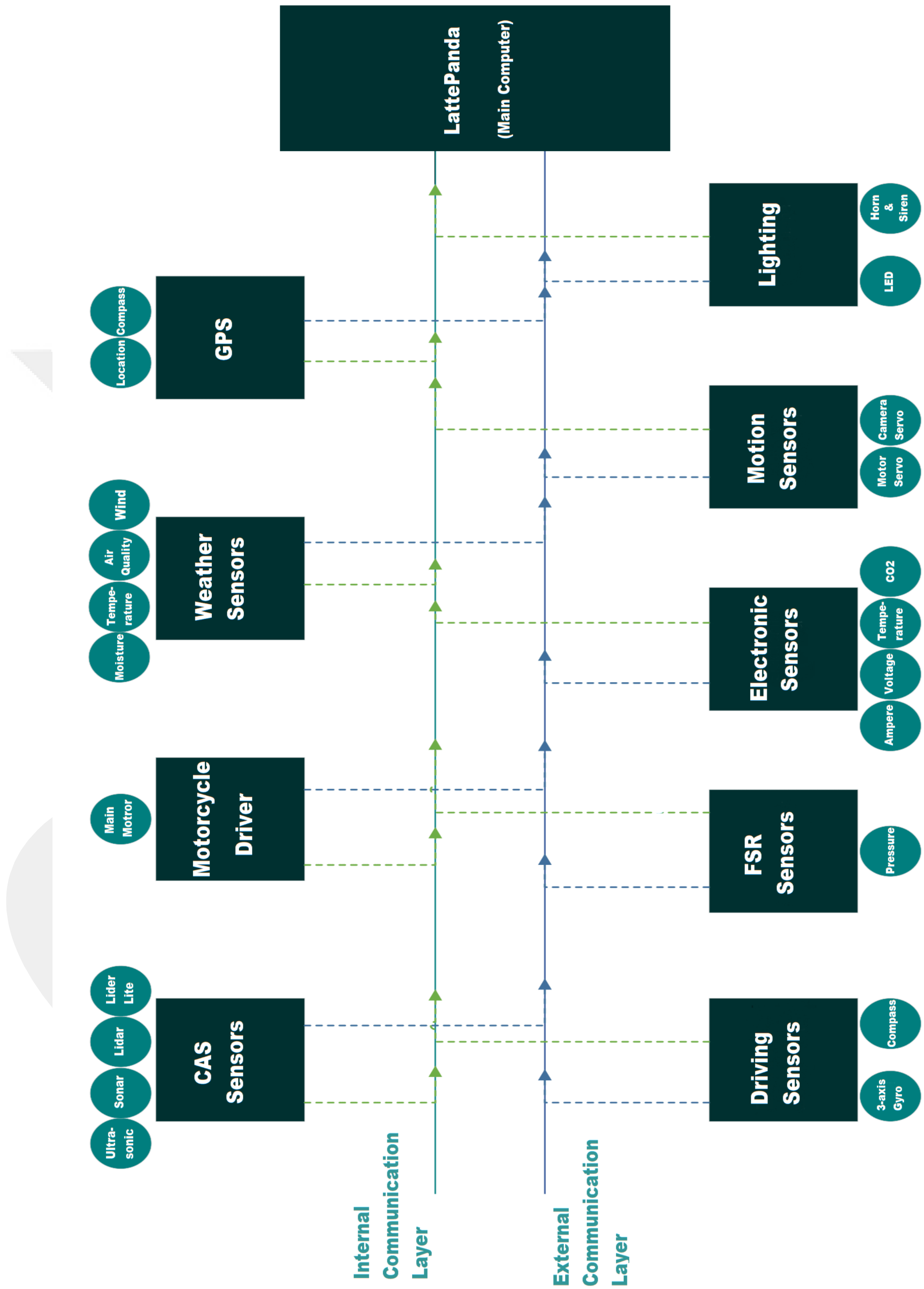

Figure 3.1: Internal and external communication layers of the sensors connections [57].

The various sensors, GNSS and more that are used to hold this operation must be functionally integrated, this is done by connecting each sensor in the breadboard and programming these sensors by using a special device called Arduino Mega. The Arduino Mega is working on control the way the sensors work according to the task required for the purpose of implementation. A several Arduinos will be used due to the quantity of

sensors that are used to collect data on the USV. This number of Arduinos that is used need to be integrated in one system. Based on that, these Arduinos are connected side by side with internal and external communication layers to an advanced device that's called Lattepanda Alpha (Figure 3.1). This smart device is the main computer that will process and transfer information of the unmanned watercraft simultaneously and frequently in real-time at operation times.

The unmanned surface vehicle that allocated for life-saving purposes is provided by a holding platform to be used by people to hold the platform when they are exposed to danger of suffocation.

The holding platform consists of a metal pipe system that surrounds the perimeter of the unmanned surface vehicle.

On the metal pipe system, the touch sensors are located and surround all the vehicle's directions. The touch sensors start to produce data from the holding area of the metal pipe system at the time when any touch operation to the holding platform occurs.

A person may hold onto this holding platform when he is in danger of suffocation. As soon as the person, who is in danger of suffocation, is attached to the holding platform, the touch sensors status is recognized when sensors are active by the unmanned watercraft and the boat realizes that a human being is holding onto it.

The main computer starts to collect and save the data of touch sensors as soon as the touch sensor that forms the holding platform begins to flow data from any position in the metal pipe system.

During operation time, the GPS system works on determine where the holding takes place and the current location of the USV.

The number of times that attachments are held with its location on the map, the location on the map when the holding takes place and the direction in which the holding area is located are recorded and transferred on the internet to a server.

On the Internet, due to the big quantity of data that generated continuously by the USV, this server, located in the cloud computing system, interprets data from the main computer on the unmanned surface vehicle (Figure 3.2).

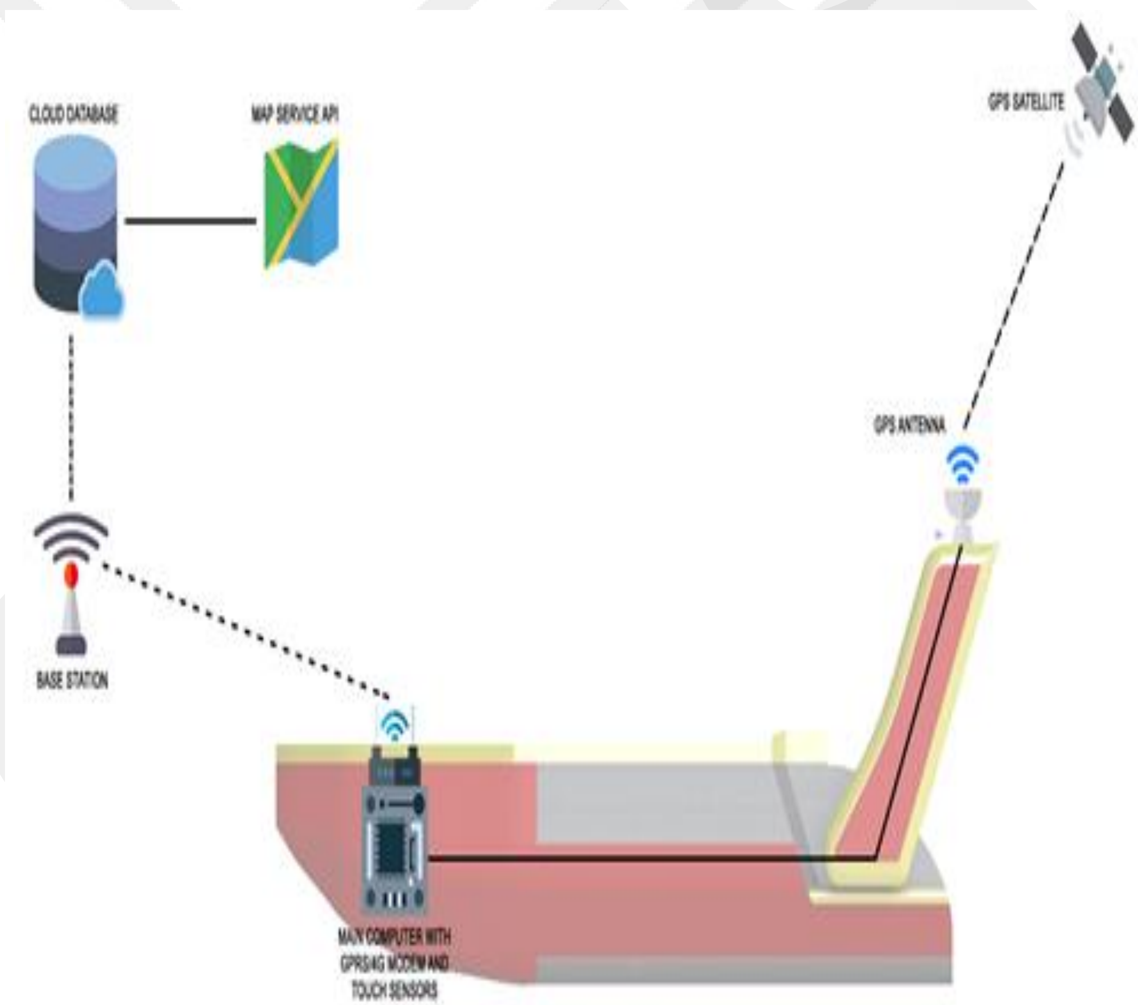

Figure 3.2: The general operation’s mechanism [57].

3.1.1 Map Creation Process

At the time when the cloud server receives the GNSS, touch sensors and time information of the unmanned surface vehicle. The processing of these data starts by cloud server on an interactive map. The location, sensors and time data, that are held to the unmanned watercraft, are regularly marked on this map. Consequently, with the multiplication of the data, an interactive drowning map is created.
When creating the map, the following data are used:

- When the grip held.
- Based on the collected the data from the touch sensor, which place on the USV that grip detection is detected.
- Number of grip operations that held.
- GPS location information when the grip held.
- The history of unmanned watercraft movements.

On the interactive digital map, the data in this information is processed continuously by the corresponding server. The data transferred to the server multiplies when each hold occurs, and the interactive map begins to hold more data. Based on received information, the server starts to generate the drowning map. This map is developed and updated in the number of people who are in danger of drowning.

During the operation times, the main computer processes and sends direction, GPS, touch and time information that is generated to the server each time and with the multiplication of the data, the map is intended to become more accurate.

First, when touch sensors run by holding it. The interactive drowning map begins to mark the location of the unmanned surface vehicle on the map, shown in Figure 3.3. Then, on the marked map point it is written as meta-data. For each new attachment this process is repeated.

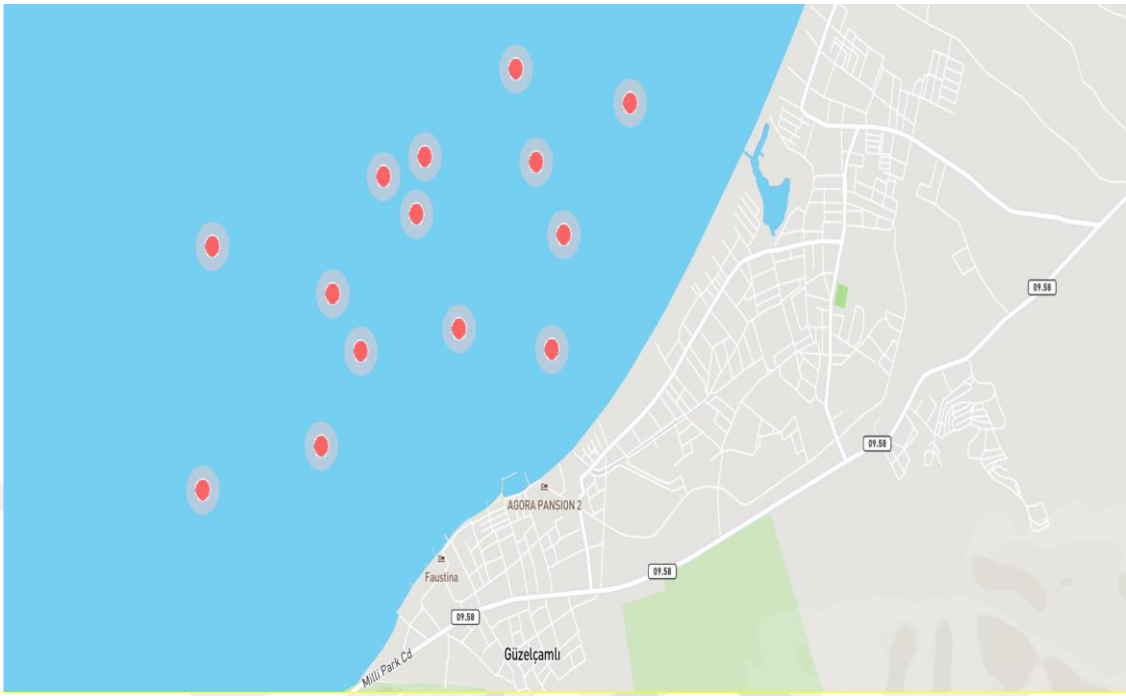

Figure 3.3: The generated landmarks on the map [57].

For information analyzing purposes, a grid system feature is added to the interactive drowning map to split the map up to squares. Based on the number of marked points per square, a dangerous zone will be determined on the map according to the number of points marked on a map if it is decreasing or increasing in a certain square.

Based on the number of points marked per square, the limited area (squares where points have marked) starts to be colored into different colors on the map layer starts from green color till red color gradients.

If a lot of location points are marked in a certain part of the coast where the unmanned watercraft is located, this area is colored with red color. If the dense location point is marked in a certain area of the coast where the unmanned watercraft is located, this area is colored with pale red color. If there are less location points in a certain area of the coast where the unmanned watercraft is located, this area is colored with orange color. If there are less location points in a certain area of the coast where the unmanned watercraft is located, this region is colored with yellow color. If less it will be colored with pale green or green for the less number. When there is no location point in a certain area of the coast where the unmanned watercraft is located, this area is has no color.

3.1.2 Grid Map System

To generate the algorithm within the grid map, the following parameters were taken as variables:

• Point (Location, Time, Direction)
• Grid (C, G)
• Colors (Green = 1~20 Range, Pale Green = 40~20 Range, Orange = 80~60 Range, Yellow = 60~40 Range, Pale Red = 100~80 Range, Red = 100 + Range)
• A GridRiskValue

Within these variables, the data point that is called Point Variable contains the operation's location, the precise direction, time and date that is held.

The data used to determine the grid square that is called Grid variable is formed by the C and G values in order to define a specific region of the map. In order to color the grids in a palette of six colors, a data is used called Colors.

For determination the color of the respective frame, the color ranges are used where they are sorted from 0 to 100. The RiskValue is the value that represents the number of points in one frame.

For the coloring of squares, this value is the main risk value to be used.

Each square is named with the name C1, G1 within the Grid map, shown in Table 3.1. Points are in the content of these frames. Within these points, the above variables are kept. The algorithm performs as follows;

1. The GridRiskValue is determined based on number of points per frame.
2. With the QuickSort algorithm, these GridRiskValue values are sorted from small to large for each grid frame.

3. The smallest GridRiskValue is marked as Green. The largest GridRiskValue is marked as Red.

4. the largest value of GridRiskValue are compared with all GridRiskValue values. Based on the ratio differences, the maximum value is determined.

5. If the largest GridRiskValue value differs from any square by 10% to 20%, these squares are marked with Pale Red.

6. If the largest GridRiskValue value differs from any square by 20% to 40%, these squares are marked with Orange.

7. If the largest GridRiskValue value differs from any square by 40% to 60%, these squares are marked with Yellow.

8. If the largest GridRiskValue value differs from any square by 60% to 80%, these squares are marked with Pale Green.

9. If the largest GridRiskValue value differs from any square by 80% to 99%, these squares are marked with Green.

10. Every hour, these valuations and comparisons are updated. As well colors and lists are updated continuously.

Table 3.1: Grid variable.

	G1	G2	G3	G4	G5
C1	G1,C1	G2,C1	G3,C1	G4,C1	G5,C1
C2	G1,C2	G2,C2	G3,C2	G4,C2	G5,C2
C3	G1,C3	G2,C3	G3,C3	G4,C3	G5,C3
C4	G1,C4	G2,C4	G3,C4	G4,C4	G5,C4
C5	G1,C5	G2,C5	G3,C5	G4,C5	G5,C5
C6	G1,C6	G2,C6	G3,C6	G4,C6	G5,C6
C7	G1,C7	G2,C7	G3,C7	G4,C7	G5,C7

3.2 Hardware Side

This section presents how to tie the sensors to the board and programs the sensors on the Arduinos. Also how to integrate more than one sensor on one board to integrate their functions will be presented as well. Fisrt the senors will be setup by using breadboard, wires, microcontroller (Arduino) and computer then it will be connected to the to the lattePanda.

3.2.1 The Touch Sensor Connection

With the touch sensor that works based on capacitive change, the finger touch can be sensed. Touch sensor has VCC pin works on voltage range from 2.0 V up to 5.5 V, also it

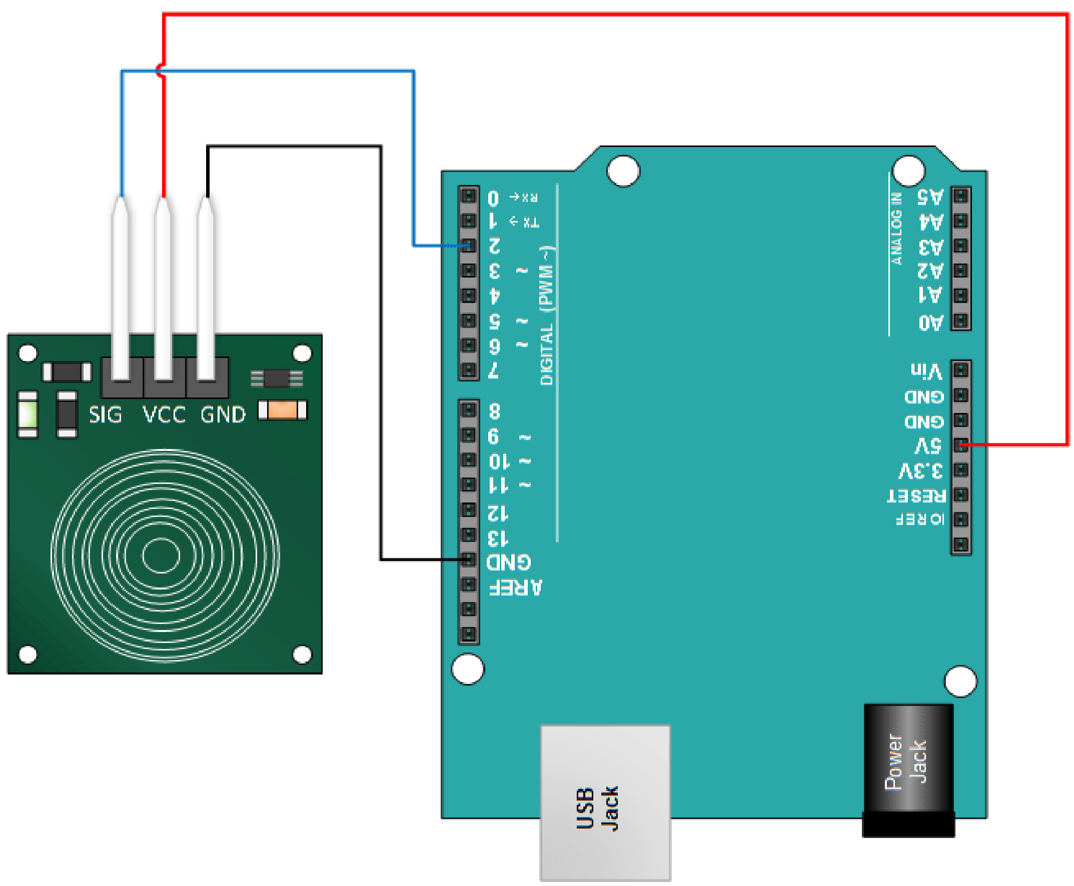

Figure 3.4: Illustration of the touch sensor connection [58].

has an low and high output in addition with the ground (GND), these accesses are used in connection of the sensor on the breadboard and Arduino.

The VCC pin of the sensor is connected to the 5 voltage pin in the Arduino in order to provide the power to the sensor. The output pin of the sensor is connected to digital I/O pin in Arduino. The ground of the sensor is connected to the GND pin in Arduino as shown in the Figure 3.4. The Arduino is connected by it's turn in the computer by using a USB cable [58].

After the connection of the hardware, software turn comes in. this is done by write the following code in the Arduino IDE, Figure 3.5.

```
int ledPin = 13; // pin for the LED

void setup() {
  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
  pinMode(ctsPin, INPUT);
}

void loop() {
  int ctsValue = digitalRead(ctsPin);
  if (ctsValue == HIGH){
    digitalWrite(ledPin, HIGH);
    Serial.println("TOUCHED");
  }
  else{
    digitalWrite(ledPin,LOW);
    Serial.println("not touched");
  }
  delay(500);

}
```

Figure 3.5: Arduino code of the touch sensor initialization [58].

3.2.2 The GNSS Module Connection

To determine the unmanned surface vehicle location and to land mark the location of people at the time they hold the boat, a TOPGNSS receiver is adopted. The GNSS model GN-8603G-H supports GPS and GLONASS satellites. It has a baud rate out of 9600, output level RS232 and refresh rate of 1 HZ. The GNSS works on voltage (VCC) range from 12 V- 24 V with red wire, the GND (ground) with a black wire. The transmitting and receiving (TX and RX) are with green and yellow wires.

To connect the TOPGNSS to the Arduino, the VCC is connected to the 5V pin of the Arduino. The ground pin is connected to any GND pin in Arduino. The RX pin of the GNSS module is connected to the pin 3 in the Arduino. The TX pin of the GNSS is connected to the pin 4 in the Arduino, Figure 3.6.

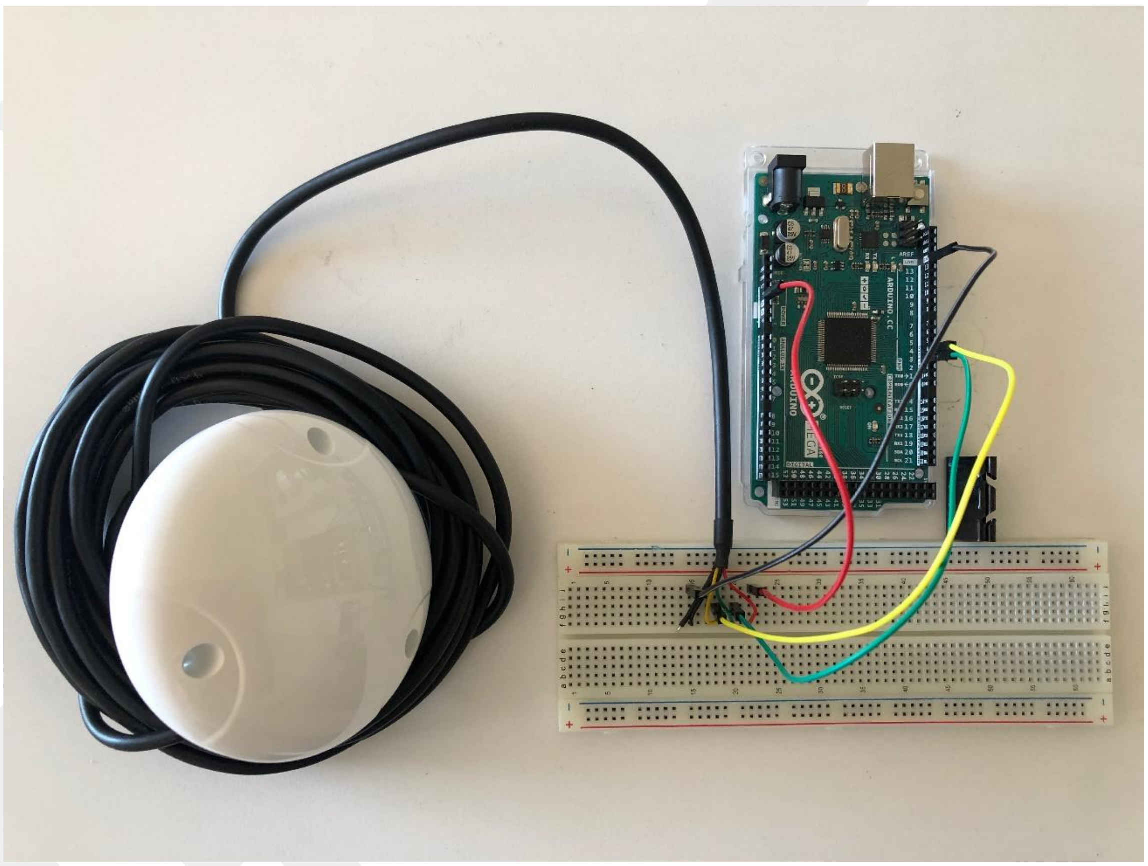

Figure 3.6: The GNSS connection.

After the connection of the hardware, software turn comes in. this is done by write the GNSS code in the Arduino IDE.

3.3 Software Side

This part highlights the way system works and the mechanism to do this. Its divide in two sections, first the mechanism system starts from the unmanned surface vehicle until the cloud system. The second half shows the mechanism system in the cloud and the drowning map system.

3.3.1 The USV - Cloud Mechanism

The first section of the mechanism system depends on the data that flowing from the sensors on the USV to the main computer. By turn on the system, the sensors (GNSS, touch sensors …etc.) start flowing data to the main computer. The GNSS and touch sensors send data continuously to the main computer. When the touch sensor status is turned on, which part of the USV that touch sensor was activated data, in addition to the

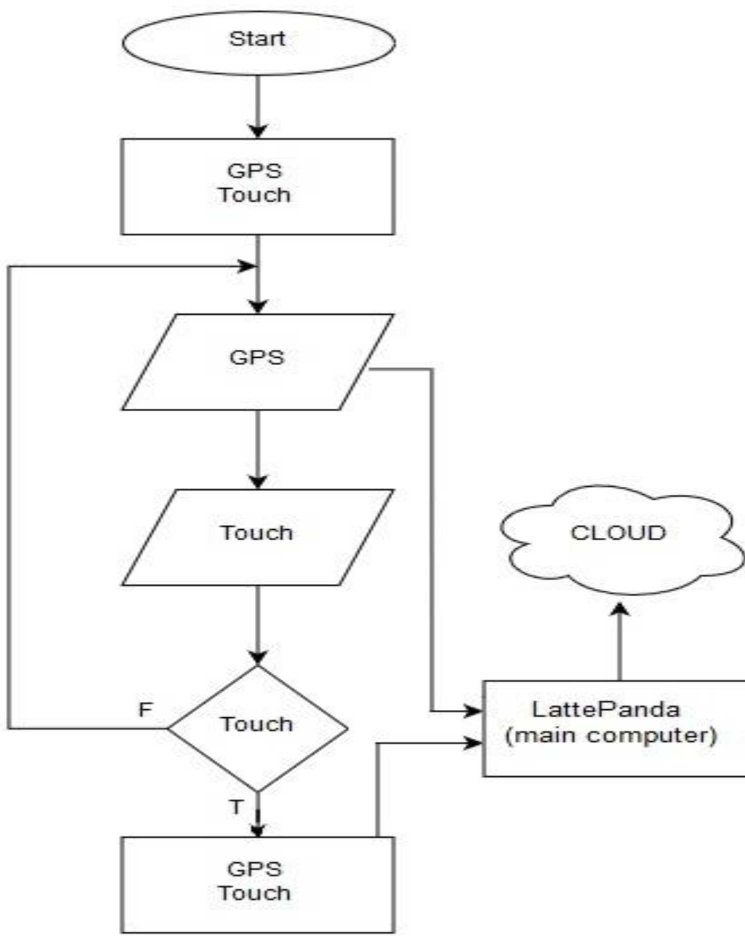

Figure 3.7: The USV - Cloud mechanism.

coordinates of the USV that is determined by the GNSS at the touch sensor activation time are sent it along with touch sensor status data to the main computer.

This information is sent by the main computer to the cloud computing in order to save this information and processing it by the drowning map system that is already located on the cloud computing, Figure 3.7.

3.3.2 The Cloud- Drowning Map Mechanism

The second section of the mechanism system starts from cloud until map interface. After the data flowed to the cloud computing through main computer, it will be saved in the database of the cloud computing. The operation held is tagged with an ID number that

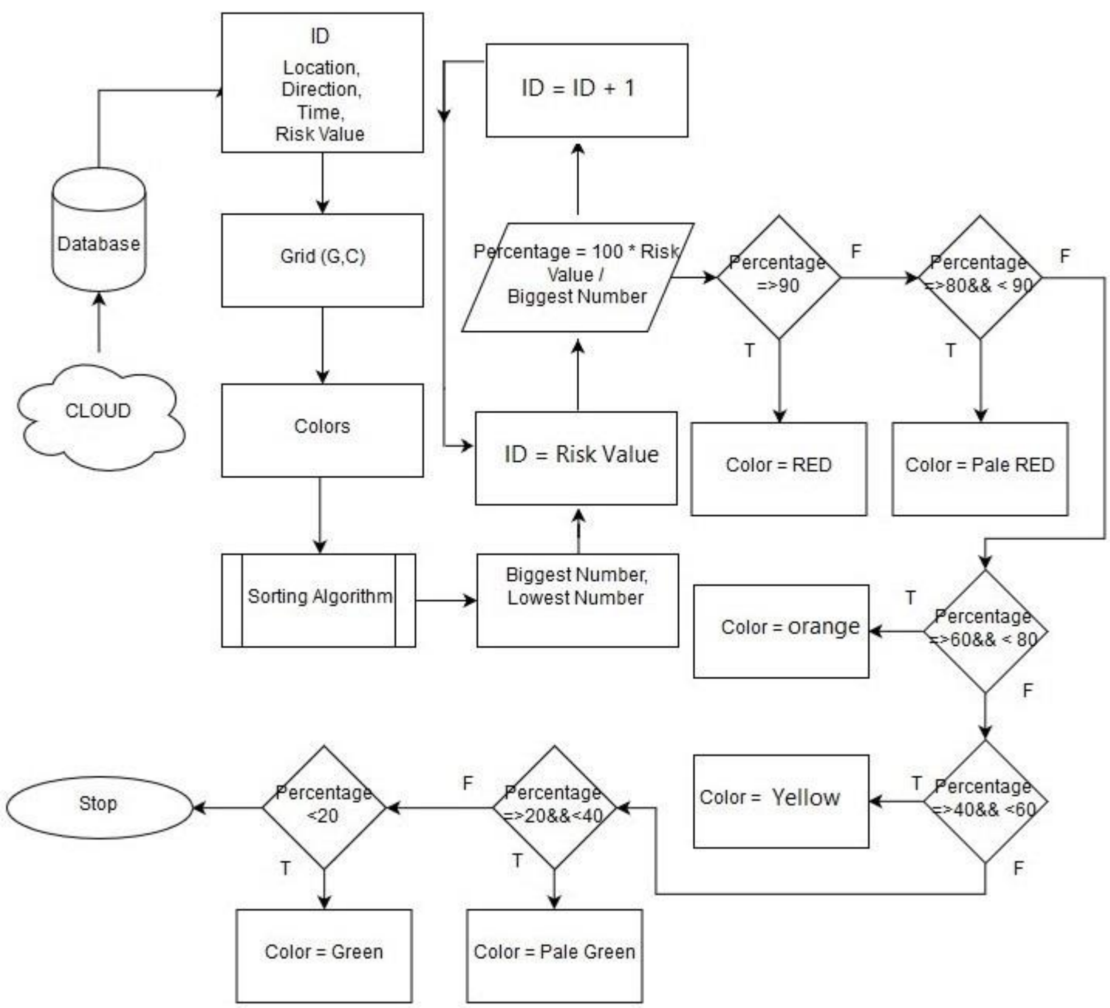

Figure 3.8: The Cloud- Drowning Map mechanism.

contains the time held the boat, risk value, location and direction. The coordinates of the grid map system are determined with the G and C. The risk values are sorted from the lowest to the biggest value with the sorting algorithm. For each grid, the dangerous level is calculated by calculate the difference percentage of the risk value of the related grid to the gird with the biggest risk value. The operation is done by multiplying the risk value of the related grid by 100 and divide the result on the biggest risk value. This is repeatedly calculated from the lowest to biggest risk value. Where each risk value is tagged with an ID number. After the difference percentage of each grid are calculated, the system starts to color the grids based on these percentages. Where the highest difference percentage is colored with green and the lowest difference percentage is colored with red. Likewise the other grids are colored with color gradients from green to the red based on the difference percentage, (Figure 3.8).

By integrate the entire USV- Cloud- Drowning Map Mechanism, the algorithm will be as shown below in Figure 3.9.

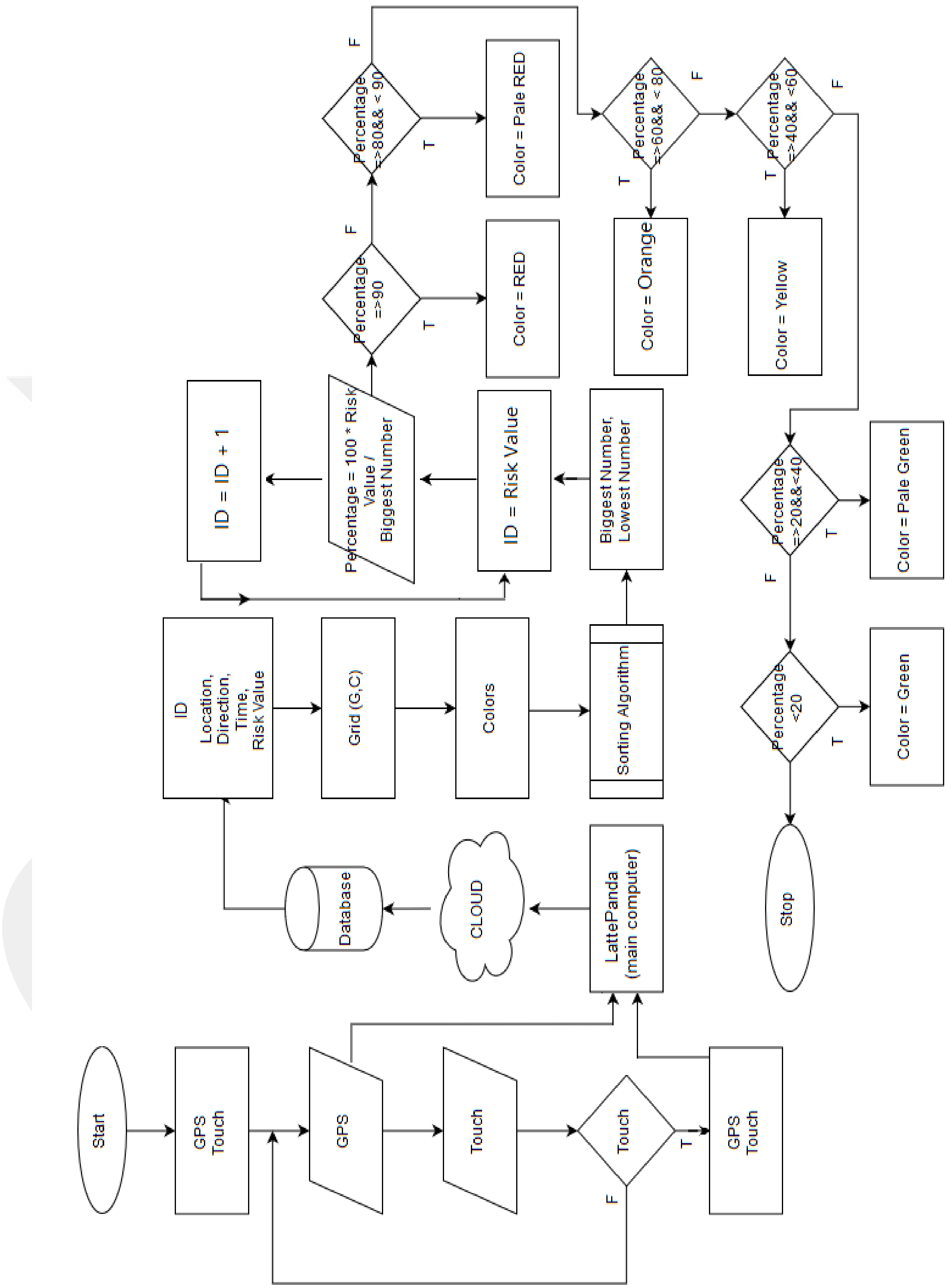

Figure 3.9: The entire USV- Cloud- Drowning Map mechanism.

CHAPTER 4

THE EXPERIMENTAL RESULTS

4.1 Results

The proposed drown map system has been tested and verified based on virtual inputs. where the attached results showed the efficiency and effectiveness level of the surveillance system for analyzing and determining the dangerous zones on the map.

The drowning map system works on two scenarios:

1) Landmark the positons of people who hold the USV on the gird map system based on GNSS and touch sensors.
2) Determining the dangerous zones with different color grades on the map based on number of landmarks per pixel.

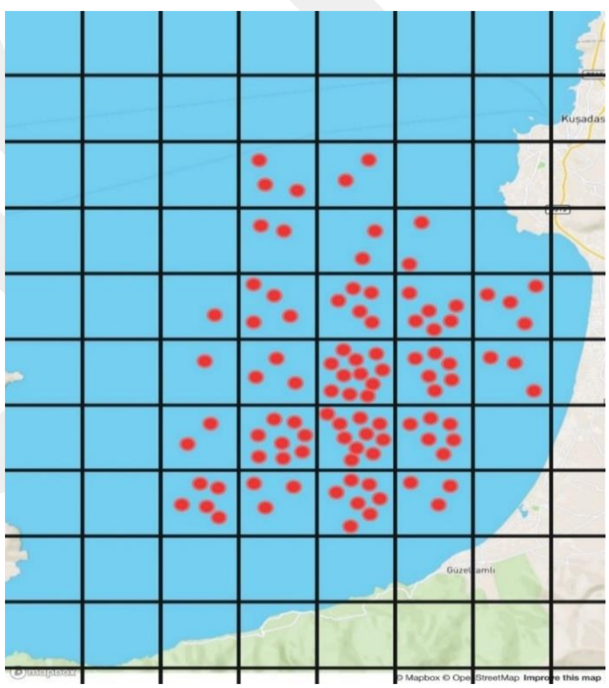

Figure 4.1: The landmarks are determined based on the touch sensors and GNSS.

In the first scenario, at the time of operation when the person touched the touch sensor it worked as an input coordinates of a landmark using the GNSS.

So the coordinates of the landmark are determined based on the touch sensor. Likewise, the coordinates of the other land marks are determined in the same way as it shown in Figure 4.1. Where there are a various number of landmarks per pixel. Based on that, the dangerous zone level is to be determined.

The second scenario, the results showed that the occupied pixel were colored with different color grades by using the proposed algorithm and based on the number of landmarks per pixel, Figure 4.2.

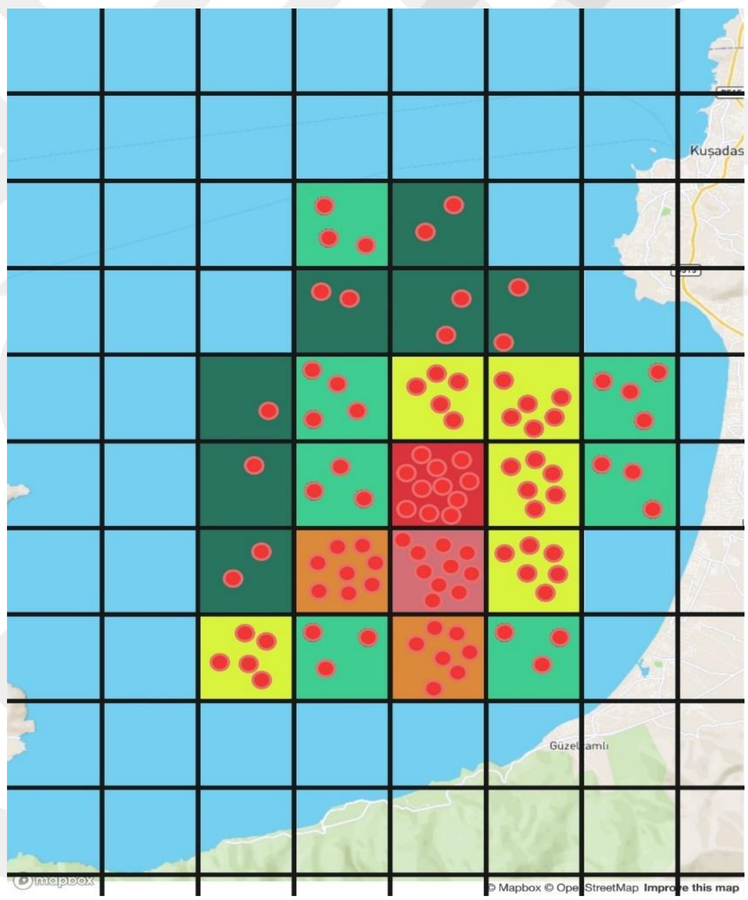

Figure 4.2: The dangerous zone level based on number of landmarks per pixel.

Where the pixels were colored as the following below:

- Without any landmarks, pixel is without color.

- with one or two landmarks, the difference percentage is between 82~91%, pixel was colored with green.
- with three or four landmarks, the difference percentage is between 64~73%, pixel was colored with light green.
- with five or six landmarks, the difference percentage is between 46~55%, pixel was colored with yellow.
- with seven or eight landmarks, the difference percentage is between 28~37%, pixel was colored with orange.
- with ten landmarks, the difference percentage is different by 10%, pixel was colored with pink or light red.
- with eleven landmarks, the pixel was colored with red.

Based on that, the last result is with the dangerous zones that have been determined on the grid map as it illustrated below in Figure 4.3.

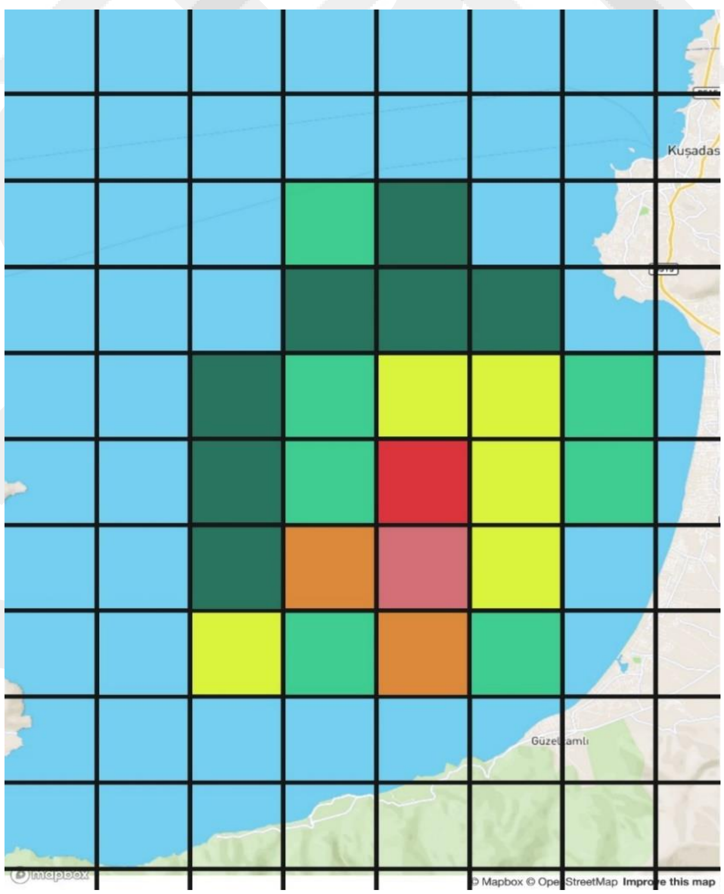

Figure 4.3: The dangerous zones.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

The thesis explained how to explore the people who exposed to drowning and how to determine the dangerous zones based on the location and number of the people who hold the USV on the map in real-time by designing and developing new algorithm. The effectiveness and capabilities of the algorithm has been tested in computer based simulations using virtual values.

As stated in the 'Introduction' Chapter, this thesis focuses on developing drowning map system using a dangerous zones determination algorithm with comprehensive optimization of multiple constraints for practical prospective drown victims exploration. By summarizing and analyzing the work, it can be concluded that this experimental has been achieved through the following aspects:

The gained results showed the advantages of the drown map system in determined the dangerous zones area on the grid map based on the number of landmarks that have been located per pixel. Where the coordinates of these landmarks have been entered randomly as virtual input by the programmer on the map. The dangerous zone levels varied between the high level of dangerous and low level. This information helps the rescue teams to improve the rescue operation in the sea in order to decrease the drowning cases, also to identify places on the map where sailing may cause potential drowning incidents. The adopted algorithm works on the determination of the dangerous zone levels based on determination the maximum GridValue and compare it with other GridValues. The difference percentage between the maximum GridValue and other values is the substantial 222factor of coloring the pixels. The error percentage of this algorithm lies in determination of the constant value of the maximum GridValue. For instance, if the maximum GridValue in one operation is five and in another operation is 12, both pixels of these values will be colored with red. So, this algorithm determines the dangerous zone levels based on the collected GridValue values per operation.

5.2 Future Work

The recommendation for the future work is to develop the algorithm of this thesis to be more advanced for determining and analyzing the dangerous zones on the map. In order to increase the response time, system development is recommended to make the system able to contact with the nearest rescue team by determining the USV position through the GNSS signals. Moreover, the dangerous zone levels can be determined by buffers rather than pixels. As well the determination of the maximum GridValue can be solved by developing the algorithm. Furthermore, it is highly recommended to add an image processing method to the system for further analysis to determine the exact number of the people who hold the boat. All the proposals mentioned above and more are recommended.

REFERENCES

[1] Larcom, J. A., & Liu, H. (2013, December). Authentication in GPS-directed mobile clouds. In *2013 IEEE Globecom Workshops (GC Wkshps)* (pp. 470-475). IEEE.

[2] http://www.terrisgps.com/gnss-gps-differences-explained/ [Accessed 24/07/2019] (20:13).

[3] http://www.terrisgps.com/gnss-gps-differences-explained/ [Accessed 02/06/2019] (01:30).

[4] https://www.semiconductorstore.com/blog/2018/GPS-vs-GNSS-What-Engineers-Need-To-Know-For-Their-Design-Symmetry-Blog/3083 [Accessed 02/06/2019] (02:00).

[5] Al-Tekreeti M., Tarıyan Özyer S., Özdemir C. and Karadağ A., Al-Dakheel S., "Unmanned Surface Vehicle with Obstacle Avoidance System", International Conference on Big data, IoT, and Cloud Computing (ICBICC), 2019, in press.

[6] Díaz, M., Martín, C., & Rubio, B. (2016). State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing. Journal of Network and Computer applications, 67, 99-117.

[7] Botta, A., De Donato, W., Persico, V., & Pescapé, A. (2016). Integration of cloud computing and internet of things: a survey. Future Generation Computer Systems, 56, 684-700.

[8] Manyika, J. (2011). Big data: The next frontier for innovation, competition, and productivity. http://www. mckinsey. com/Insights/MGI/Research/Technology_and_Innovation/Big_data_The_next_frontier_for_innovation.

[9] Ali, W. B. (2016). Big data-driven smart policing: big data-based patrol car dispatching. Journal of Geotechnical and Transportation Engineering, 1(2).

[10] Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqa, A., & Yaqoob, I. (2017). Big IoT data analytics: architecture, opportunities, and open research challenges. *IEEE Access*, *5*, 5247-5261.

[11] https://www.unhcr.org/europe-emergency.html [Accessed 08/06/2019] (07:30).

[12] https://data2.unhcr.org/en/situations/mediterranean#_ga=2.242048325.659918465.1559990315-1899383023.1559990315 [Accessed 08/06/2019] (07:45).

[13] Schofield, R. T., Wilde, G. A., & Murphy, R. R. (2018, August). Potential Field Implementation for Move-to-Victim Behavior for a Lifeguard Assistant Unmanned Surface Vehicle. In *2018 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)* (pp. 1-2). IEEE.

[14] Cao, H., Guo, Z., Gu, Y., & Zhou, J. (2018, October). Design and Implementation of Unmanned Surface Vehicle for Water Quality Monitoring. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* (pp. 1574-1577). IEEE.

[15] Ko, N. Y., Choi, H. T., Lee, C. M., & Moon, Y. S. (2016, September). Navigation of unmanned surface vehicle and detection of GPS abnormality by fusing multiple sensor measurements. In *OCEANS 2016 MTS/IEEE Monterey* (pp. 1-5). IEEE.

[16] Akulakrishna, P. K., Lakshmi, J., & Nandy, S. K. (2014, December). Efficient storage of big-data for real-time gps applications. In *2014 IEEE Fourth International Conference on Big Data and Cloud Computing* (pp. 1-8). IEEE.

[17] Corfield, S. J., & Young, J. M. (2006). Unmanned surface vehicles-game changing technology for naval operations. *IEE Control Engineering Series*, *69*, 311.

[18] Motwani, A. (2012). A survey of uninhabited surface vehicles. *Marine and Industrial Dynamic Analysis, School of Marine Science and Engineering, Plymouth University, Tech. Rep*.

[19] Song, R. (2018). *Path Planning and Collision Avoidance of Unmanned Surface Vehicles in the Marine Environment* (Doctoral dissertation, UCL (University College London)).

[20] Manley, J. E. (2008, September). Unmanned surface vehicles, 15 years of development. In *OCEANS 2008* (pp. 1-4). Ieee.

[21] Manley, J. E. (2016, September). Unmanned Maritime Vehicles, 20 years of commercial and technical evolution. In *OCEANS 2016 MTS/IEEE Monterey* (pp. 1-6). IEEE.

[22] Liu, Z., Zhang, Y., Yu, X., & Yuan, C. (2016). Unmanned surface vehicles: An overview of developments and challenges. *Annual Reviews in Control*, *41*, 71-93.

[23] Manley, J. E. (1997, October). Development of the autonomous surface craft" ACES". In *Oceans' 97. MTS/IEEE Conference Proceedings* (Vol. 2, pp. 827-832). IEEE.

[24] Manley, J. E., Marsh, A., Cornforth, W., & Wiseman, C. (2000, September). Evolution of the autonomous surface craft AutoCat. In *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No. 00CH37158)* (Vol. 1, pp. 403-408). IEEE.

[25] Curcio, J., Leonard, J., & Patrikalakis, A. (2005, September). SCOUT-a low cost autonomous surface platform for research in cooperative autonomy. In *Proceedings of OCEANS 2005 MTS/IEEE* (pp. 725-729). IEEE.

[26] Pascoal, A., Oliveira, P., Silvestre, C., Sebastião, L., Rufino, M., Barroso, V., ... & Ryan, A. (2000, September). Robotic ocean vehicles for marine science applications: the european asimov project. In *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No. 00CH37158)* (Vol. 1, pp. 409-415). IEEE.

[27] Martins, A. A., Almeida, J., Silva, A. F., & Pereira, F. L. (2006). Vision-based autonomous surface vehicle docking manoeuvre. In *MCMC'2006-7th Conference on Manoeuvring and Control of Marine Craft*.

[28] Ferreira, H., Martins, A., Dias, A., Almeida, C., Almeida, J. M., & Silva, E. P. (2007). Roaz autonomous surface vehicle design and implementation. *Robótica Controlo, Automação, instrumentação*.

[29] Naeem, W., Xu, T., Sutton, R., & Tiano, A. (2008). The design of a navigation, guidance, and control system for an unmanned surface vehicle for

environmental monitoring. *Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment*, *222*(2), 67-79.

[30] Sutton, R., Sharma, S., & Xao, T. (2011). Adaptive navigation systems for an unmanned surface vehicle. *Journal of Marine Engineering & Technology*, *10*(3), 3-20.

[31] Kaizu, Y., Iio, M., Yamada, H., & Noguchi, N. (2011). Development of unmanned airboat for water-quality mapping. *Biosystems engineering*, *109*(4), 338-347.

[32] Liu, Y., Noguchi, N., & Yusa, T. (2014). Development of an unmanned surface vehicle platform for autonomous navigation in paddy field. *IFAC Proceedings Volumes*, *47*(3), 11553-11558.

[33] Caccia, M., Bono, R., Bruzzone, G., Spirandelli, E., Veruggio, G., Stortini, A. M., & Capodaglio, G. (2005). Sampling sea surfaces with SESAMO: an autonomous craft for the study of sea-air interactions. *IEEE robotics & automation magazine*, *12*(3), 95-105.

[34] Li, C., & Weeks, E. (2009). Measurements of a small scale eddy at a tidal inlet using an unmanned automated boat. *Journal of Marine Systems*, *75*(1-2), 150-162.

[35] Twichell, D., Andrews, B., Edmiston, L., & Stevenson, B. (2007). Geophysical mapping of oyster habitats in a shallow estuary: Apalachicola Bay, Florida. USGS.

[36] Botta, A., De Donato, W., Persico, V., & Pescapé, A. (2014, August). On the integration of cloud computing and internet of things. In 2014 International Conference on Future Internet of Things and Cloud (pp. 23-30). IEEE.

[37] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

[38] https://www.kbmanage.com/concept/cloud-computing-in-supply-chain [Accessed 30/07/2019] (9:51).

[39] Subashini, S., & Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. Journal of network and computer applications, 34(1), 1-11.

[40] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. Journal of internet services and applications, 1(1), 7-18.

[41] Dukaric, R., & Juric, M. B. (2013). Towards a unified taxonomy and architecture of cloud frameworks. Future Generation Computer Systems, 29(5), 1196-1210.

[42] https://apprenda.com/library/paas/iaas-paas-saas-explained-compared/ [Accessed 20/06/2019] (18:13).

[43] https://www.uniprint.net/en/7-types-cloud-computing-structures/ [Accessed 30/07/2019] (9:30).

[44] Subirana, J. S., Zornoza, J. J., & Hernández-Pajares, M. (2013). GNSS Data processing. Volume 1: Fundamentals and Algorithms. *ESA Communications ESTEC, PO Box, 299*, 2200.

[45] http://www.terrisgps.com/gnss-gps-differences-explained/ [Accessed 24/07/2019] (20:13).

[46] Hofmann-Wellenhof, B., Lichtenegger, H., & Wasle, E. (2007). GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more. Springer Science & Business Media.

[47] Stephenson, S. (2016). *Automotive applications of high precision GNSS* (Doctoral dissertation, University of Nottingham).

[48] http://what-when-how.com/gps/gps-positioning-modes-part-1/ [Accessed 23/07/2019] (15:13).

[49] https://nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/ModernSurveyingTech/lecture4/4_4_GPS_positioning.htm [Accessed 23/07/2019] (16:13).

[50] https://www.e-education.psu.edu/geog862/node/1845 [Accessed 23/07/2019] (14:13).

[51] https://www.dfrobot.com/product-1729.html [Accessed 24/07/2019] (16:20).

[52] http://mosaic.cnfolio.com/ENG500Coursework201718CW1IRFMFV/files.xml?action=download&file=LattePanda%20Alpha%20version%202%20%281%29.pdf [Accessed 24/07/2019] (18:46).
[53] https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf [Accessed 28/07/2019] (18:06).
[54] http://www.alselectro.com/arduino-mega-2560.html [Accessed 28/07/2019] (19:13).
[55] https://www.uploadarchief.net/files/download/ttp223_module.pdf [Accessed 19/08/2019] (11:06).
[56] https://www.dx.com/p/digital-capacitive-touch-sensor-switch-module-for-arduino-blue-golden-2043660#.XV2BYugzZPY [Accessed 21/08/2019] (20:30).
[57] Al-Dakheel S., Tarıyan Özyer S., Özdemir C. and Karadağ A., Al-Tekreeti M., "Unmanned Surface Vehicle with Drown Map System", International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2019, in press.
[58] http://henrysbench.capnfatz.com/henrys-bench/arduino-sensors-and-input/catalex-ttp223b-arduino-capacitive-touch-sensor-tutorial/ [Accessed 22/08/2019] (13:07).