MINISTERE DE L'EQUIPEMENT, DU TRANSPORT, DE LA LOGISTIQUE ET DE L'EAU
*****
HIGHER INSTITUTE OF MARITIME STUDIES

**Department:** NAVAL ENGINEERING

# UNDERGRADUATE THESIS PROJECT

**To Obtain the Diploma of Bachelor Degree in Marine Engineering**

**THEME :**

# SMART OBSTACLE DETECTION SYSTEM FOR SHIPS NAVIGATION ASSISTANCE BASED ON ARTIFICIAL INTELLIGENCE

**Defended on : 09 / 06 / 2021 By :**

- E/O **NDIAYE Mouhamadou Bouchra I.**
- E/O **TOURAY Omar**

**Members of the Jury :**

| | |
|---|---|
| - Pr. Briouig Mohamed | President |
| - Pr. Hachim Abdelilah | Member |
| - Pr. Ech-cheikh Hamid | Member |
| - Pr. Douraid Abdessamad | Supervisor ISEM |
| - Pr. Bacha Abdelkabir | Supervisor ISEM |

# *DEDICATIONS*

*To our dear parents, for all their sacrifices, their love, their tenderness, their support and prayers throughout our studies.*

*To our brothers, sisters and friends for their constant encouragement and moral support.*

*To all our family for their support throughout our school career.*

*May this work be the fulfillment of your wishes and the fruit of your unfailing support.*

*Thank you for always being there for us.*

# *ACKNOWLEDGEMENTS*

*We first thank the Almighty and Merciful God, who has given us the strength and patience to do this modest work.*

*This project would not have been possible without the conscious intervention of some people. We would like to thank them here.*

*We would like to express our sincere gratitude to Mr. Douraid Abdessamad, Mr. Echcheikh Hamid and Mr. Bacha Abdelkabir as our supervisors and mentors for this project. Despite their very busy schedules, we were able to count on their advice and availability throughout the completion of this work.*

*Our gratitude also goes to the Higher Institute of Maritime Studies (ISEM) and all of its academic staff for having provided invaluable efforts during these years to create an environment conducive to the quality of our studies.*

*Finally, we thank all those who in one way or another, have contributed to the realization of this work.*

# ABSTRACT

The maritime transport lanes are becoming more and more dense and the total number of ships is constantly growing. Globally, the safety situation at sea is progressively getting more complicated and various types of ship accidents have occurred including those at the time of docking of the ships. Improving maritime transport and ensuring the safety of maritime navigation is an important direction of research in the field of maritime navigation.

The integration of new technologies and approaches in the navigation systems of ships can considerably improve their safety, and the tools of artificial intelligence present themselves as a solution filling the gaps of conventional radars. The association of artificial intelligence on one side and cameras as an external sensor on the other has given satisfaction in several areas.

The idea of this project is to build a system able to detect fixed obstacles as well as mobile ones. Using images captured in real time by cameras attached to the ship's exterior hull, the machine learning tool developed can promptly report obstacles that could put the ship at risk.


***Keywords:*** *Artificial Intelligence, Machine Learning, Camera, Obstacle Detection, Neural Network, Deep Learning, Conventional radar approach, Sensor fusion*

# TABLE OF CONTENTS

**DEDICATIONS**……………………………………………………………………...1

**ACKNOWLEDGEMENTS**…………………………………………………………2

**ABSTRACT**………………………………………………………………………3

**FIGURES**…………………………………………………………………………9

**ACRONYMS**……………………………………………………………………..10

**INTRODUCTION**………………………………………………………………..11

**PART 1: GENERAL OVERVIEW OF THE CONVENTIONAL RADAR USED FOR OBSTACLE DETECTION IN SHIPS**

**SECTION A: DESCRIPTION OF THE CONVENTIONAL RADAR USED FOR SHIP NAVIGATION**

**A – 1 HISTORICAL OVERVIEW** ........ 13

**A – 2 RADAR PRINCIPLES** ........ 13

A -2.1 BASIC WORKING PRINCIPLE OF THE CONVENTIONAL RADAR SYSTEM........ 13

A-2.2 GENERAL OPERATIONAL PRINCIPLES OF THE MARINE RADAR SYSTEM........ 15

A-2.2.1 THE TRANSMITTER........ *15*

A-2.2.2 THE RADAR ANTENNA ........ *16*

A-2.2.3 THE RECEIVER........ *16*

A-2.2.4 SIGNAL AND DATA PROCESSORS........ *17*

A-2.2.5 THE DISPLAY........ *18*

A-2.3 RADAR MEASUREMENT ........ 19

A-2.3.1 HOW TO CALCULATE THE DISTANCE FROM A TARGET**?** (RANGE MEASUREMENT)........ *19*

A-2.3.2 BEARING MEASUREMENT ........ 20

**A-3 THE RADAR / ARPA SYSTEM** ........ 20

A-3.1 DEFINITION OF ARPA ........ 20

A-3.2 IMO PERFORMANCE STANDARDS FOR ARPA ........ 21

**SECTION B: LIMITS OF THE CONVENTIONAL RADAR SYSTEM IN OBSTACLE DETECTION**

**B-1 RADAR PERFORMANCE**........ 22

**B-2. FACTORS AFFECTING RADAR PERFORMANCE**........ 22

B-2.1 RECEIVER NOISE ..... 22
B-2.2 TARGET SIZE ..... 22
B-2.3 CLUTTER ECHOES ..... 23
B-2.4 INTERFERENCE ..... 23
**B-3 RADAR ERRORS** ..... 23
B-3.1 BEAMWIDTH ERROR ..... 23
B-3.2 INDEX ERROR ..... 23
B-3.3 ATTENUATION ERROR ..... 23
B-3.4 INDIRECT WAVE ERROR ..... 24
B-3.5 MULTIPLE ECHOES ..... 24
**B-4 LIMITATION OF THE RADAR** ..... 24
B-4.1 THE PROBLEM OF TARGET DISCRIMINATION ..... 24
B-4.2 LACKING OF THE ABILITY TO CLASSIFY DETECTED TARGETS ..... 24
B-4.3 BLIND RANGE OF NO DETECTION ..... 24
B-4.4 FALSE AND UNWANTED RADAR RESPONSES ..... 25
**PART 2: UTILISATION OF ARTIFICIAL INTELLIGENCE AND CAMERAS TECHNOLOGIES FOR OBSTACLE DETECTION IN SHIP NAVIGATION**
**SECTION A: BRIEF INTRODUCTION TO ARTIFICIAL INTELLIGENCE, MACHINE LEARNING AND DEEP LEARNING**
**A – 1 GENERAL OVERVIEW OF AI** ..... 26
A-1.1 AI DEFINITION ..... 26
A-1.2 AI HISTORY ..... 27
A-1.3 CLASSIFICATION OF AI ..... 28
A-1.3.1 STAGES OF ARTIFICIAL INTELLIGENCE ..... *29*
A-1.3.2 BRANCHES OF ARTIFICIAL INTELLIGENCE ..... *30*
A-1.3.3 RELATIONSHIP BETWEEN AI, MACHINE LEARNING AND DEEP LEARNING ..... *32*
**A-2 DEEP LEARNING AND NEURAL NETWORKS** ..... 33
A-2.1 NEURAL NETWORK DEFINITION ..... 33
A-2.2 DEEP LEARNING DEFINITION ..... 34
**A-3 TYPES OF DEEP LEARNING ALGORITHMS** ..... 35
A-3.1 THE MULTILAYER PERCEPTRON (MLP) ..... 35
A-3.2 THE CONVOLUTIONAL NEURAL NETWORKS (CNNs) ..... 36
A-3.3 THE RECURRENT NEURAL NETWORKS (RNNs) ..... 37

A-3.4 THE GENERATIVE ADVERSARIAL NETWORKS (GANs) .......... 37
A-3.5 THE MODULAR NEURAL NETWORKS (MNNs) .......... 37

**SECTION B: OBJECT DETECTION**

**B –1 DEFINITION OF OBJECT DETECTION, RECOGNITION AND TRACKING WITH DEEP LEARNING** .......... 38
B-1.1 OBJECT DETECTION AND RECOGNITION .......... 38
B-1.2 OBJECT TRACKING .......... 40
**B–2 WORKING PRINCIPLE OF OBJECT DETECTION BASED ON DEEP LEARNING** .......... 41
B-2.1 REGION PROPOSAL BASED .......... 41
B-2.2 REGRESSION BASED ALGORITHMS .......... 42
B-2.2.2 SSD MOBILENET .......... *42*
B-2.2.2 YOLO .......... *42*
**B–3 AN OVERVIEW OF VISUAL CAMERA SENSORS AND COMPUTING POWER REQUIREMENTS FOR OBJECT DETECTION WITH DEEP LEARNING** .......... 43
B-3.1 AN OVERVIEW OF VISUAL CAMERA SENSORS FOR OBJECT DETECTION .......... 43
B-3.1.1 STEREO CAMERA .......... *43*
B-3.1.2 MONOCULAR CAMERA .......... *43*
B-3.1.3 RGB-D CAMERA .......... *44*
B-3.1.4 OMNIDIRECTIONAL CAMERA .......... *44*
B-3.1.5 FISHEYE CAMERA .......... *44*
B-3.1.6 EVENT-BASED CAMERA .......... *44*
B–3.2 COMPUTING POWER REQUIREMENTS FOR OBJECT DETECTION WITH DEEP LEARNING .......... 44

**SECTION C: PROPOSITION OF A REAL SOLUTION AND ITS ADVANTAGES ON SHIP NAVIGATION SAFETY**

**C-1 PRESENTATION OF THE PROPOSED ARCHITECTURE OF THE REAL SOLUTION** .......... 45
C.1.1 ARCHITECTURE WITH CAMERA-ONLY AS EXTERNAL SENSORS .......... 45
C-1.1.1 COMBINATION OF LONG AND SHORT-RANGE CAMERAS .......... *45*
C-1.1.2 WIRED OR WIRELESS COMMUNICATION .......... *46*
C-1.1.3 ONBOARD COMPUTER PROCESSORS .......... *46*
C-1.1.4 AI SOFTWARE .......... *46*
C-1.1.5 THE MONITOR .......... *47*
C-1.2 SENSOR FUSION ARCHITECTURE .......... 48
C-1.2.1 THE RADAR .......... *48*

C-1.2.2 AIS – AUTOMATIC IDENTIFICATION SYSTEM ........ *48*

C-1.2.3 ECDIS – ELECTRONIC CHART DISPLAY INFORMATION SYSTEM ........ *49*

C-1.2.3 GNSS – GLOBAL NAVIGATION SATELLITE SYSTEM ........ *49*

**C-2 ADVANTAGES OF THE SOLUTION IN SHIP NAVIGATION SAFETY** ........ 49

## PART 3: PROTOTYPING OF THE SMART OBSTACLE DETECTION SYSTEM

### SECTION A: SYSTEM'S ANALYSIS & FEASIBILITY STUDIES

**A-1 SYSTEM'S ANALYSIS** ........ 52

A-1.1 SYSTEM'S REQUIREMENTS OF THE PROJECT ........ 52

A-1.1.1 HARDWARE REQUIREMENTS ........ *52*

A-1.1.2 SOFTWARE REQUIREMENTS ........ *52*

A-1.1.3 PROJECT MANAGEMENT ........ *53*

A-1.2 FUNCTIONAL ANALYSIS ........ 54

A-1.2.1 FUNCTIONAL ANALYSIS OF REQUIREMENTS ........ *54*

A-1.2.1 TECHNICAL FUNCTIONAL ANALYSIS ........ *56*

**A-2 FEASIBILITY STUDIES** ........ 57

A-2.1 LEGAL FEASIBILITY ........ 57

A-2.2 TECHNICAL FEASIBILITY ........ 58

A-2.3 ECONOMICAL FEASIBILITY ........ 58

### SECTION B: DESCRIPTION OF THE PRO¨POSED SYSTEM ARCHITECTURE

**B-1 DESCRIPTION OF THE COMPONENTS OF THE SYSTEM** ........ 62

B-1.1 DESCRIPTION OF THE RASPBERRY PI BOARD ........ 62

B-1.2 COMPONENTS REQUIRED TO CONFIGURE THE RASPBERRY-PI ........ 64

B-1.3 THE CAMERA ........ 65

B-1.4 ULTRASONIC SENSOR ........ 65

B-1.5 BREADBOARD, LEDS, RESISTORS AND JUMPER WIRES ........ 66

**B-2 SUMMARY OF THE PROPOSED ARCHITECTURE** ........ 67

### SECTION C: PRESENTATION OF THE DATASET AND THE NEURAL NETWORK ALGORITHM DEVELOPED

**C-1 PRESENTATION OF THE DATASET** ........ 68

C-1.1 DEFINITION OF DATASET AND WHY IT IS USEFUL FOR THE PROTOTYPE ........ 68

C-1.2 PRESENTATION OF THE DATASET FOR THE PROTOTYPE ........ 68

**C-2 THE NEURAL NETWORK ALGORITHM DEVELOPED FOR THE PROTOTYPE** ........ 69

A-2.1 NEURAL NETWORK ALGORITHM TRAINED WITH THE COCO DATASET ..................... 69

C-2.2 NEURAL NETWORK DEVELOPED FOR THE SHIP-ONLY DATASET ................................ 70

**SECTION D: SIMULATION AND SOLUTION PROTOTYPING**

**D-1 CONFIGURATION OF THE RASPBERRY PI** ............................................................. 73

D-1.1 INSTALLATION OF THE OPERATING SYSTEM (OS) IN THE SD-CARD ........................... 73

D-1.2 CONNECTION OF THE RASPBERRY PI TO THE COMPUTER ............................................ 75

**D-2 IMPLEMENTATION OF THE NEURAL NETWORK ALGORITHM IN THE RASPBERRY PI** ............................................................. 77

D-2.1 UPDATE OF THE RASBERRY PI ............................................................. 77

D-2.2 INSTALLATION OF TENSORFLOW ............................................................. 78

D-2.3 INSTALLATION OF OPENCV ............................................................. 78

D-2.4 COMPILATION AND INSTALLATION OF PROTOBUF ............................................ 79

D-2.5 SET UP OF THE TENSORFLOW DIRECTORY STRUCTURE ............................................ 79

D-2.6 DOWNLOAD OF THE SSDLITE_MOBILENET_V2 ............................................ 80

D-2.7 EXECUTION OF THE PROGRAM IN THE PI TERMINAL ............................................ 80

**D-3 CONNECTION OF THE COMPONENTS OF THE SYSTEM** ............................................ 82

**D-4 CONFIGURATION AND EXECUTION OF THE ULTRASONIC SENSOR PROGRAM ALONG WITH THE LEDS** ............................................................. 82

**D-5 EMULATION OF THE SMART OBSTACLE DETECTION SYSTEM** ..................................... 84

**CONCLUSION** ............................................................. 90

**REFERENCES** ............................................................. 91

# FIGURES :

Figure 1: Simplified RADAR block diagram [2] ........ 14
Figure 2: Block diagram of a basic marine radar system [1]........ 15
Figure 3: horizontal and the vertical plane of the beam transmitted by the antenna [8] ........ 16
Figure 4: Marine Radar Screen [10]........ 18
Figure 5: The echo principal [1]........ 19
Figure 6: Representation of the horizontal beamwidth [4]........ 20
Figure 7: Self-Driving Car [21]........ 29
Figure 8: AI at the stage of AGI [22] ........ 30
Figure 9: Sophia, the humanoid robot, first unveiled to the world in 2016, is a good example of AI in robotics [24]. ........ 32
Figure 10: A simple representation of a biological neuron [26]........ 34
Figure 11: representation of a simple artificial neural network: x1, x2, and x3 are the features of the input data; w1, w2 and w3 represent the weights: The weights are parameters that have to be trained using the input features and their labels and y is the output label [27]. ........ 34
Figure 12: Distinction between the representation of a simple neural network and deep learning [28]. ........ 35
Figure 13: Example of a multi-layer perceptron with two hidden layers, one input and one output layer [29]. ........ 36
Figure 14: Representation of the CNN architecture recognizing an image [30] ........ 37
Figure 15: Representation of some detectable classes[32]. ........ 39
Figure 16: technique of object localization (where the coordinates of the bounding box are defined) [33]. 39
Figure 17: Example of Object Detection [34]. ........ 40
Figure 18: Steps involved in multi-object tracking [36]. ........ 41
Figure 19: Example of ships detected with AI software (using deep learning algorithm) [39]. ........ 47
Figure 20: A typical image of the environment that can be displayed by a monitor [39]. ........ 48
Figure 21: Bull Chart [authors] ........ 54
Figure 22: Octopus Diagram [authors]........ 55
Figure 23: FAST diagram [authors] ........ 56
Figure 24: First level SADT diagram [authors] ........ 57
Figure 25: Raspberry pi board........ 62
Figure 26: Raspberry Pi 3 model b........ 64
Figure 27: The USB webcam connected to the raspberry pi through the USB port........ 65
Figure 28: HC-SR04 Ultrasonic Sensor distance sensor ........ 66
Figure 29: LEDs, resistors and jumper wires in the breadboard ........ 66
Figure 30: Uploading the dataset to the Google Collaboratory notebook [authors]........ 71
Figure 31: unzipping the uploaded file and import all the dependencies [authors]........ 71
Figure 32: Visualization of the train images [authors] ........ 72
Figure 33: importation of the pre-trained model used [authors] ........ 72
Figure 34: Training of the model with the ship dataset [authors] ........ 72
Figure 35 : SD Memory Card Formatter interface ........ 74

Figure 36: Image of the Raspbian operating system to download ........ 74
Figure 37: Interface of the ETCHER application ........ 75
Figure 38: Text file configured for the connection of the pi to the mobile Hotspot named AndroidAP12 ... 75
Figure 39: Interface of the Advanced Ip Scanner software ........ 76
Figure 40: Interface of the PUTTY software ........ 76
Figure 41: Interface of the VNC Viewer ........ 77
Figure 42: Pi Desktop ........ 77
Figure 43: Detection of a cup with an accuracy of 86% ........ 80
Figure 44: Detection of the Authors and their respective laptops by the Algorithm ........ 81
Figure 45: Connection of the components of the system ........ 82

# ACRONYMES :

| | |
|---|---|
| **AGI** | Artificial General Intelligence |
| **AI** | Artificial Intelligence |
| **AIS** | Automatic Identification System |
| **ANI** | Artificial Narrow Intelligence |
| **ANN** | Artificial Neural Network |
| **ARPA** | Automatic Radar plotting Aids |
| **ASI** | Artificial Super Intelligence |
| **CF** | Constraint function |
| **CPA** | Closest Point of Approach |
| **CNN** | Convolutional Neural Network |
| **COLREG** | International Regulations for Preventing Collisions at Sea |
| **CF** | Constraint Function |
| **DNN** | Deep Neural Network |
| **ECDIS** | Electronic Chart Display Information System |
| **FPGA** | Field Programmable Gate Array |
| **GNN** | Generative Adversarial Network |
| **GNSS** | Global Navigation Satellite System |
| **IMO** | International Maritime Organization |
| **ML** | Machine Learning |
| **MLP** | Multilayer Perceptron |
| **MNN** | Modular Neural Network |
| **OpenCV** | Open source Computer Vision |
| **PF** | Principal Function |
| **Pi** | Raspberry pi |
| **PPI** | Plan Position Indicator |
| **RADAR** | Radio Detection and Ranging |
| **RNN** | Recurrent Neural Network |
| **SSD** | Single Shot Director |
| **TF** | TensorFlow |
| **YOLO** | You Only See Once |

# INTRODUCTION

Since the first industrial revolution, the maritime industry has been advancing enormously by harnessing the development of new technologies. However, safety problems especially collision risks are always well-known for posing great challenges.

Although many strict regulations and rules have been put into place by the COLREGs (International Regulations for Preventing Collisions at Sea) to minimize ship collisions, more and more ships are deployed to the sea and the risks of collision have never been higher. This increase in maritime traffic complexity has to be proportional to the adoption of more sophisticated and efficient navigation tools for maritime vessels. Today, conventional radars along with ARPAs (Automatic Radar Plotting Aids) remain the ultimate tool available for seafarers to make manoeuvring decisions. These traditional technologies, we believe, have not kept up with the complexity of today's crowded seas and a more efficient and assuring system can be built and used by harnessing the power of Artificial Intelligence along with camera fusion technologies.

In fact, cameras have been successfully employed in many robotics applications, and they can be used to detect close-range obstacles in a radar's blind zone at a relatively high sampling rate. They can be used to obtain precise relative bearing and range of close-range obstacles in the radar's blind zone, also cameras have a relatively high angular resolution, and thus they can enhance the detectability of short- or mid-range targets. The integration of cameras in the ship navigation system can improve the performance of target detection compared to that of the conventional radar-only-based approach.

The goal of the project is to propose and build a prototype of a smart obstacle detection system for ship navigation assistance that is based on deep learning (Artificial Intelligence software) and camera sensor fusion. The system should be able to detect obstacles and evaluate the maritime environment in real-time and give warnings regarding collision avoidance, thus helping the crew to make the necessary maneuvers.

In the following lines, we will first deal with a general overview of the marine conventional radar and its limitations. Afterwards, we will get into the details of Artificial Intelligence and cameras technologies and then propose a real solution for the system. Finally, we will analyze and develop a prototype for the smart obstacle detection system.

# PART 1: GENERAL OVERVIEW OF THE CONVENTIONAL RADAR USED FOR OBSTACLE DETECTION IN SHIPS

# SECTION A: DESCRIPTION OF THE CONVENTIONAL RADAR USED FOR SHIP NAVIGATION

## A – 1 HISTORICAL OVERVIEW

The word RADAR is an acronym derived from the words Radio Detection and Ranging. In 1886, It was demonstrated that radio waves could be reflected from metallic objects by the scientist Heinrich Hertz. In 1904 a German engineer, Christian Hülsmeyer, obtained a patent in several countries for a radio wave device capable of detecting ships, but it aroused little enthusiasm because of its very limited range. In 1922, Marconi drew attention to the work of Hertz and proposed in principle what we know today as marine radar. Although radar was used to determine the height of the ionosphere in the mid-1920s. In the 1930s there was much simultaneous but independent development of radar techniques in Britain, Germany, France and America. Radar first went to sea in a war- ship in 1937 and by 1939 considerable improvement in performance had been achieved. By 1944 naval radar had made an appearance on merchant ships and from about the end of the war, the growth of civil marine radar began. Progressively it was refined to meet the needs of peacetime navigation and collision avoidance. While the civil marine radars of today may, in size, appearance and versatility, differ markedly from their ancestors of the 1940s, the basic data that they offer, namely target range and bearing, are determined by exploiting the same fundamental principles unveiled so long ago [1].

## A – 2 RADAR PRINCIPLES

### A -2.1 BASIC WORKING PRINCIPLE OF THE CONVENTIONAL RADAR SYSTEM

Radar is an electromagnetic based detection system that works by radiating electromagnetic waves from a transmitter via an antenna and then studying the echo or the reflected waves. The interaction with the target involves some of the energy being reflected from itself, some being absorbed and others being transmitted through. The signal which is reflected back towards the antenna gets received by the receiver through the duplexer. The duplexer is a device that isolates the receiver from the transmitter while permitting them to share a common antenna. The reflected electromagnetic signal (echo) is used to determine both the distance and the bearing of the target. During the process, the receiver does not only detect, but also amplifies and transforms the received signals into a video format.

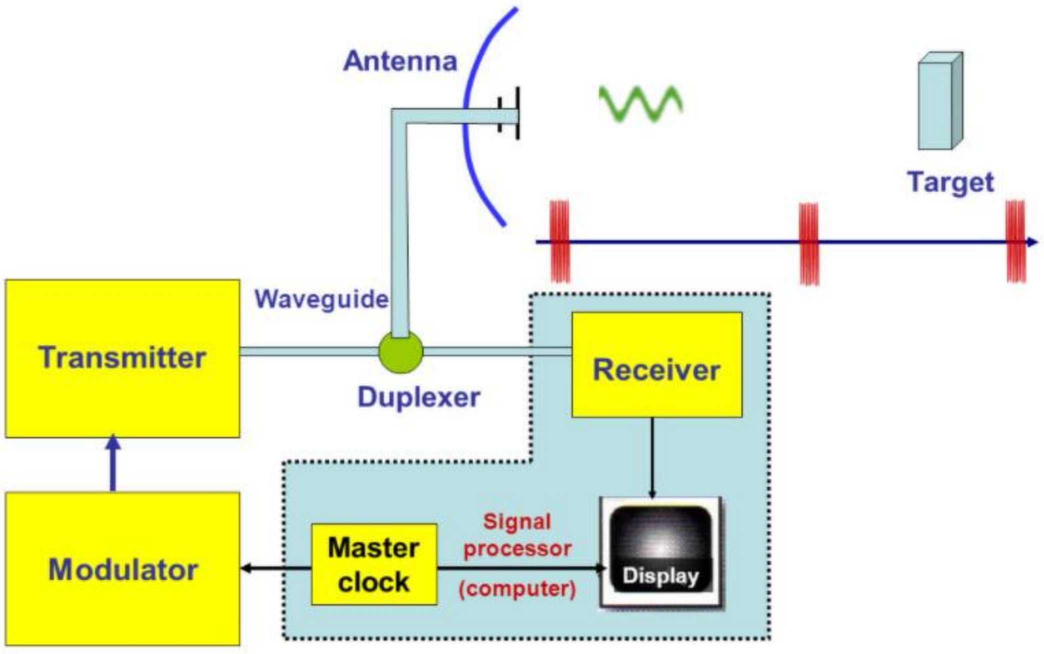



**Figure 1: Simplified RADAR block diagram [2]**

## A-2.2 GENERAL OPERATIONAL PRINCIPLES OF THE MARINE RADAR SYSTEM

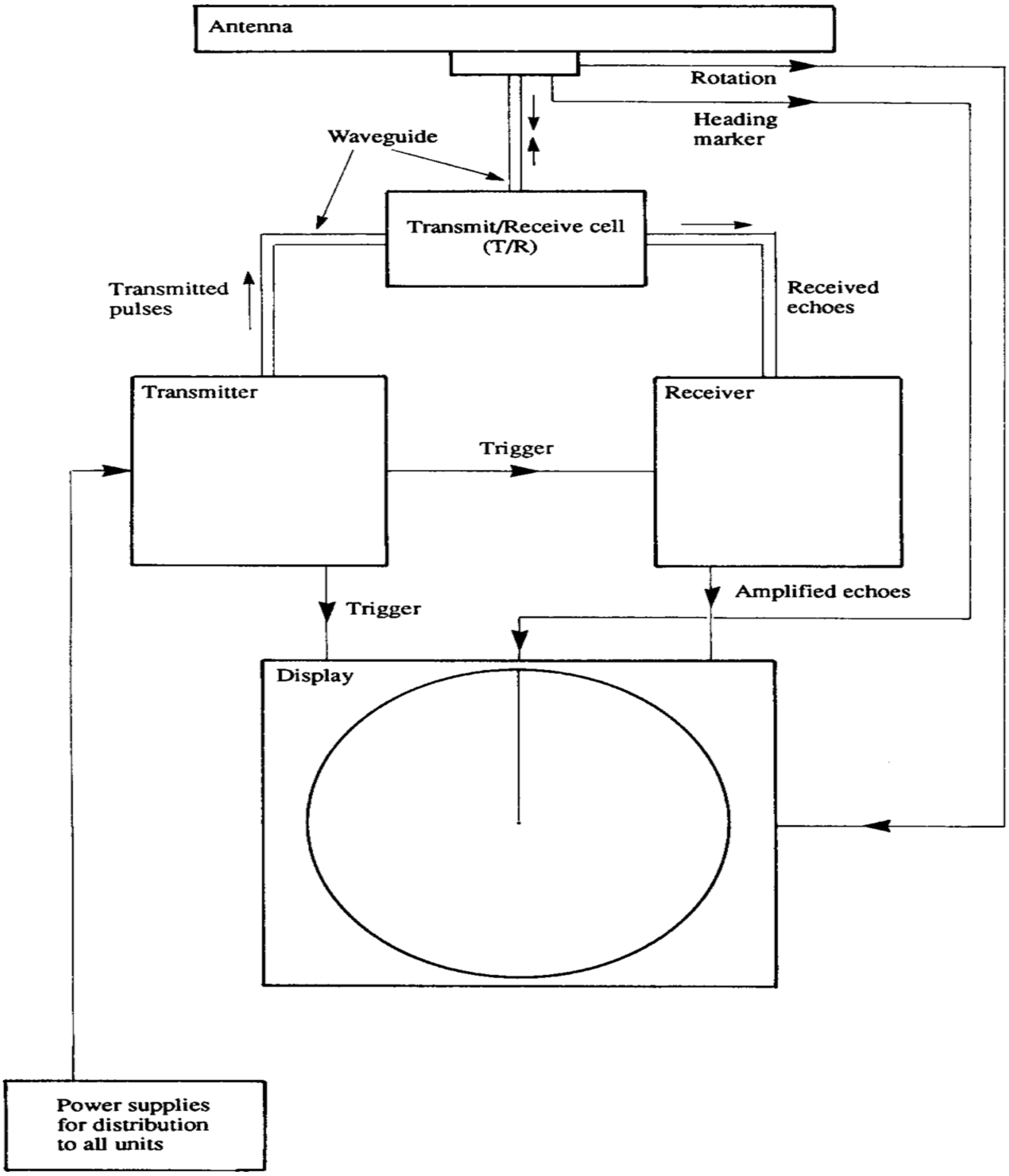



**Figure 2**: **Block diagram of a basic marine radar system [1]**

### *A-2.2.1 The Transmitter*

The transmitter is an electronic device used to generate electromagnetic waves of the radar system. In this sense, the function of the transmitter is to generate pulses of electromagnetic energy having the correct repetition frequency, length, shape, power and radio frequency. The pulses normally travel to the antenna through a hollow copper tubing which has a precisely machined rectangular or circular cross-section and is known as waveguide.

In the block diagram, a further line is shown connecting the transmitter to the receiver which carries the trigger pulse – a pulse that is used to initiate sea clutter suppression. Sea clutter is the name given to the echoes which are returned by the sea waves in the vicinity of the observing vessel.

*(clutter: is a term used for unwanted echoes in electronic systems.)*

## A-2.2.2 The Radar Antenna

Also called Scanner or Aerial, the antenna is an electronic component through which microwave pulses are transmitted and received. Its construction defines the power distribution of the radar beam in both the horizontal and the vertical planes. In order to achieve the required directional characteristic, the horizontal limits must be narrow. By contrast, the beam is wide in a vertical sense in order to maintain adequate performance when a vessel is rolling (or pitching) in a seaway. The IMO Performance Standards set out certain range performance requirements and these must be achieved when the vessel is rolling or pitching up to ±10 degrees. While in theory one might contemplate using some form of gyro stabilization to maintain the beam in a horizontal plane, in practice the standard has the effect of defining a minimum vertical beam width of 20 degrees. The scanner is rotated continuously and automatically in a clockwise direction (when viewed from above) in order to achieve the desired 360 degrees of azimuth coverage. [3]

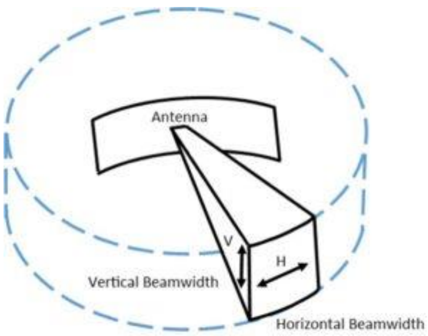



**Figure 3: horizontal and the vertical plane of the beam transmitted by the antenna [8]**

*(Beam: is a line of energy, radiation, or particles sent in a particular direction;*

*Gyro stabilization: is a process that tries to keep the ship by preventing it from rolling [13])*

## A-2.2.3 The Receiver

The radar is an electronic device used to detect the microwave pulse that is reflected by the area being imaged by the radar. So, the main function of the receiver is to amplify the very weak echoes intercepted by the aerial so as to generate pulses whose form and power will produce a visible response on the screen of a cathode ray tube (or provide a suitable input for digital storage). It will be noticed that because a single aerial is used for transmission and reception, the waveguide is common to both transmitter and receiver. It would thus appear that the powerful pulses generated by the transmitter might be able to pass directly into the receiver.

The receiver is protected from this by a device known as a transmit/receive switch (or T/R cell) which is situated in the waveguide immediately before the input to the receiver. The T/R cell blocks the input to the receiver during transmission.

In effect, there are thus two inputs to the receiver, the received signals from the aerial and the trigger pulse to initiate sea clutter suppression. The output line from the receiver carries the amplified signals to the display.

*(Trigger pulse: a pulse that starts a cycle of operation. Also known as tripping pulse.)*

### *A-2.2.4 Signal and Data Processors*

The signal processor is the part of the receiver that extracts the desired target signal from unwanted clutter. It is not unusual for these undesired reflections to be much larger than desired target echoes, in some cases more than one million times larger. Large clutter echoes from stationary objects can be separated from small moving target echoes by noting the Doppler frequency shift produced by the moving targets. Most signal processing is performed digitally with computer technology. Digital processing has significant capabilities in signal processing not previously available with analog methods [3].

*(Doppler shift: is an apparent change in frequency due to the relative motion of two objects.[9])*

## A-2.2.5 The Display

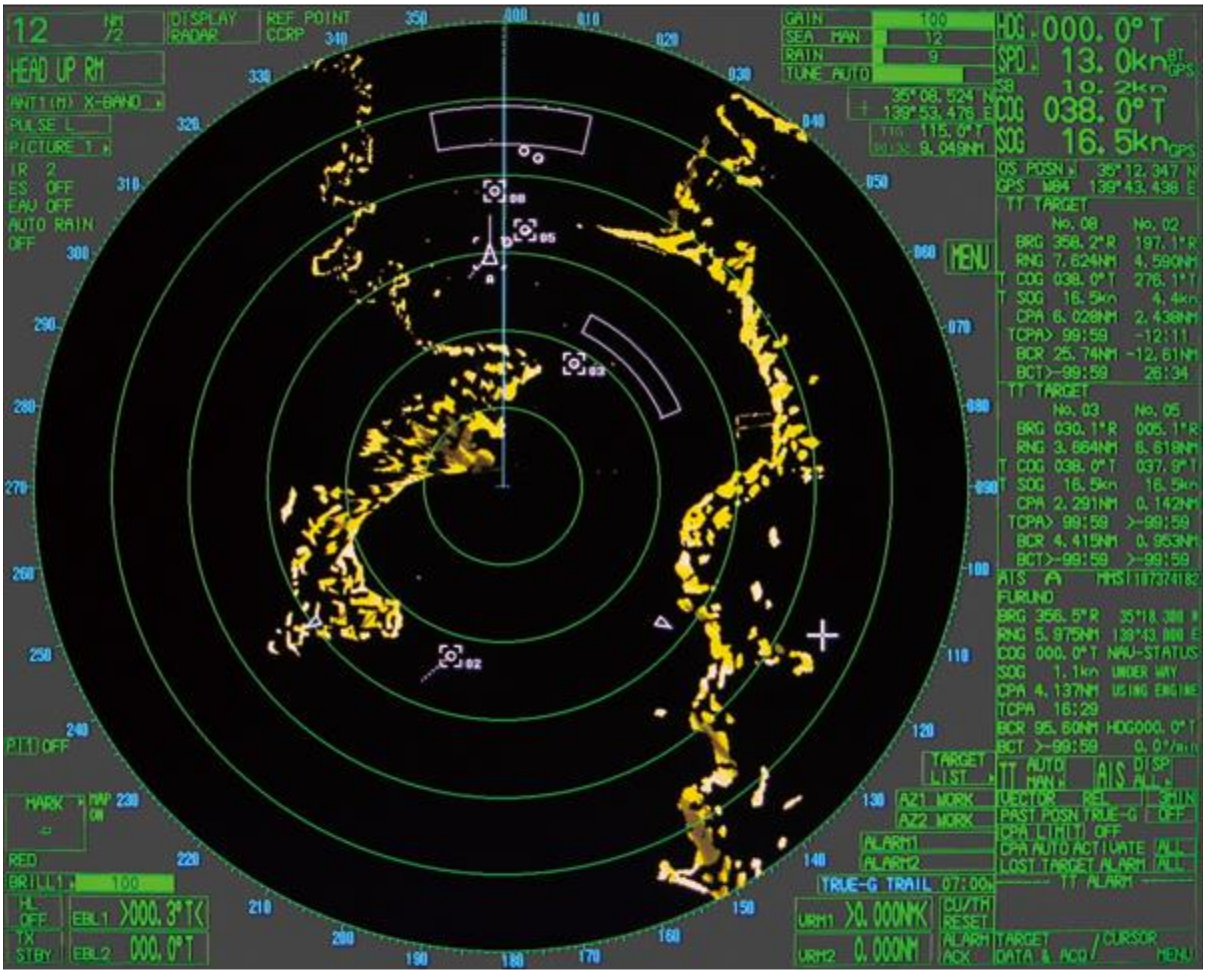



**Figure 4: Marine Radar Screen [10]**

The prime function of the display is to indicate the presence of detectable objects by generating on the screen of the cathode ray tube a visible response (a picture) whose angular and radial position with respect to the heading line and the origin of the trace are representative of the bearing and range respectively, at which the corresponding target lies. The cathode ray tube (CRT) is a glass screen across which travels a very small spot of light. The speed of this travel can be accurately controlled at values which allow the spot to transit the screen in as little as a few microseconds. At such speeds it moves literally 'faster than the eye can see' and hence appears as a line rather than a spot. The CRT can be used to perform an electronic stopwatch function by arranging that the time taken for the spot to cross the screen is the same as the time taken for a radar pulse to make the two-way journey to a target at a chosen range.[3]

Types of radar output displays:
The A-scan display, radial-scan Plan Position Indicator display (PPI) and the Raster-Scan Monitor.

## A-2.3 RADAR MEASUREMENT

### *A-2.3.1 How to calculate the Distance from a target? (Range Measurement)*

When calculating a distance (range) between the radar and an object, it must be taken account that the time T measured between the emission of a pulse and the reception of its echo is the round trip of this pulse since it is rebounded by this object. Knowing that the reception of the corresponding echo depends on the speed of the pulse and the distance, and also the elapsed time can be measured, hence the distance between the target producing the echo can be determined.

We can suppose that:

D = the distance travelled by the pulse (meter)

d = the range of the target (meter)

T = the elapsed time (microseconds)

C = the speed of radio waves (meter/microseconds) which is approximately 300 000 000 m/s

Then $D = C \times T$ (1)

Or $D = 2d$ (2)

Hence $d = (C \times T) / 2$ (3)

The minimum detection range that must be achieved to ensure compliance with the IMO Performance Standards for Navigational Radar Equipment is 40m. These Performance Standards were revised under Res. MSC.192(79) and adopted on 6 December 2004 [2]. This shows that targets too close to the ship may not be detected by the radar as they are unlikely to be within the radar range.

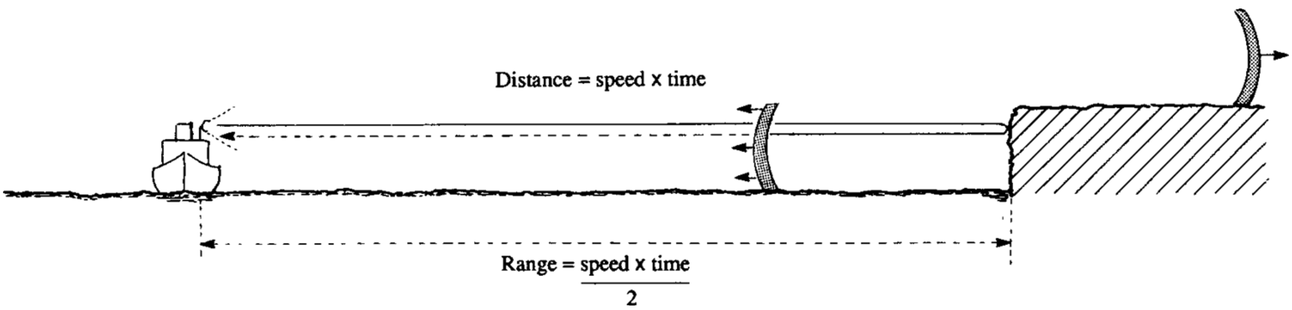



**Figure 5: The echo principal [1]**

*(IMO: International Maritime Organization*

*Res. MSC [ Resolution MSC (Maritime Safety Committee)]:* ***ADOPTION OF THE REVISED PERFORMANCE STANDARDS FOR RADAR EQUIPMENT (adopted on 6 December 2004) [12])***

### *A-2.3.2 Bearing Measurement*

In a marine radar system, a single antenna is used for both transmission and reception. It is designed in such a way as to focus the transmitted energy in a beam which is very narrow in the horizontal plane. The angle within which the energy is constrained is called the horizontal beamwidth. It must have a value of not more than 2.5 degrees if it is to comply with IMO Performance Standards for Navigational Radar Equipment. There is an exception for high speed craft in that the horizontal beamwidth can be up to 4 degrees for S-band radar only [1]. The reception property of the antenna is such that it will only detect energy which has returned from within the angular limits of the horizontal beamwidth. It follows from the directional characteristic of the antenna that only those targets which lie in the direction of the beam will appear on the trace. Thus, a single trace represents the ranges of targets lying along a specific line of bearing.

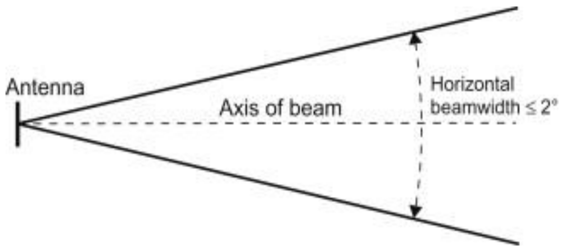



**Figure 6: Representation of the horizontal beamwidth [4]**

*( S-band radar: a part of the microwave band of the electromagnetic spectrum covering frequencies from 2 to 4 gigahertz (GHz). It is used by airport surveillance radar for air traffic control, weather radar, surface ship radar, and some communications satellites . [11])*

# A-3 THE RADAR / ARPA SYSTEM

## A-3.1 DEFINITION OF ARPA

The availability of low-cost microprocessors and the development of advanced computer technology during the 1970s and 1980s have made it possible to apply computer techniques to improve commercial marine radar systems. Radar manufacturers used this technology to create the Automatic Radar Plotting Aids (ARPAs). ARPAs are computer assisted radar data processing systems which generate predictive vectors and other ship movement information. An ARPA assesses the risk of collision, and enables operators to see proposed maneuvers by their own ship.

While many different models of ARPAs are available, the following functions are what is provided:

- True or relative motion radar presentation
- Automatic acquisition of targets plus manual acquisition
- Digital read-out of acquired targets which provides course, speed, range, bearing, closest point of approach (CPA) and time to CPA.
- The ability to display collision assessment information directly on the PPI (Plan Position Indicator), using vectors (true or relative) or a graphical Predicted Area of Danger display (PAD).
- The ability to perform trial maneuvers including, course changes, speed changes and combined course/speed changes.
- Automatic ground stabilization for navigation purposes.

ARPA processes radar information much more rapidly than a raw conventional radar but is still subject to the same limitations. ARPA data is only accurate as the data that comes from inputs such as the gyro and speed log. Over the past decade there have been considerable developments in ARPA design, but perhaps the more significant changes have been seen in the design of basic radar systems: ARPA features are now almost entirely integrated with the radar display – which means that neither can be treated in isolation.[5]

*(Speed Log: is a device that displays the speed of the ship)*

### *A-3.2 IMO Performance Standards For ARPA*

The IMO has set out certain standards amending the Standard Convention for Safety of Life at Sea (SOLAS) requirements regarding the carrying of the ARPA.

The primary function of ARPAs can be summarized under the following statement found under the IMO performance standards. It states a requirement of ARPA "*in order to improve the standards of collision avoidance at sea: reduce the workload of observers by enabling them to automatically obtain information so that they can perform as well with multiple targets as they can by manually plotting a single target".* As we can see from this statement, the principal advantages of ARPA are a reduction in workload of bridge personnel and fuller and quicker information on the selected targets [5].

# SECTION B: LIMITS OF THE CONVENTIONAL RADAR SYSTEM IN OBSTACLE DETECTION

As seen above, the RADAR is mainly used for detecting the objects and finding their location. However, this conventional radar system faces certain limitations regarding obstacle detection.

Above all, knowing how to evaluate the performance of this device is essential.

## B-1 RADAR PERFORMANCE

The performance of a radar can be judged by the following:

- The maximum range at which it can see a target of a specified size.
- The accuracy of its measurement of the target location in range and bearing (angle).
- Its ability to distinguish one target from another.
- Its ability to detect the desired target echo when masked by the clutter echoes
- Its ability to recognize the type of target
- Its availability (ability to operate when needed), reliability and maintainability.

## B-2 FACTORS AFFECTING RADAR PERFORMANCE

### B-2.1 RECEIVER NOISE

The sensitivity of a radar receiver is determined by the unavoidable noise that appears at its input. At microwave radar frequencies, the noise that limits detectability is usually generated by the receiver itself (i.e., by the random motion of electrons at the input of the receiver) rather than by an external noise that enters the receiver via the antenna.

### B-2.2 TARGET SIZE

The size of a target as seen by the radar is not always related to the physical size of the object. The measure of the target size as observed by radar is called the radar cross section and is given in units of area (square meters). It is possible for two targets with the same physical cross-sectional area to differ considerably in radar size, or radar cross-section.

### B-2.3 CLUTTER ECHOES

Echoes from land, sea, rain, snow, hail, birds, insects, and meteors are a nuisance to radars that want to detect ships, or other similar targets. Clutter echoes can seriously limit the capability of a radar system; thus, a significant part of radar design is devoted to minimize the effects of clutter without reducing the echoes from desired targets. The Doppler frequency shift is the usual means by which moving targets are distinguished from the clutter of stationary objects. Detection of targets in rain is less of a problem at the lower frequencies, since the radar echo from rain decreases rapidly with decreasing frequency and the average cross section of a ship or any sea target is relatively independent of frequency in the microwave region (likewise in the radio wave region).

### B-2.4 INTERFERENCE

Signals from nearby radars and other transmitters can be strong enough to enter a radar receiver and produce false responses. Well-trained operators are not often deceived by interference, though they may find it harmful as it can mislead them. Interference is not as easily ignored by automatic detection and tracking systems. However, some methods are usually needed to recognize and remove interference pulses before they enter the automatic detector and tracker of a radar.

## B-3 RADAR ERRORS

### B-3.1 BEAMWIDTH ERROR

When the radar beam from a ship moves away, the width of the beam tends to widen. This causes distortion of objects being detected. This distortion error increases as the vessel moves further away from the target.

### B-3.2 INDEX ERROR

This is the difference between the actual range between two points on a map and the range detected by the radar. This error can be observed when the vessels seat abeam between two points.

### B-3.3 ATTENUATION ERROR

Attenuation is caused by the absorption and subsequent scattering of the beam energy as it is transferred through the atmosphere. This usually leads to a significant reduction in the strength of the echo. Attenuation is much more common in instances where there are high frequencies and short wavelengths.

### B-3.4 INDIRECT WAVE ERROR

When a radar beam is emitted from its own ship, it's supposed to travel in a straight line directly to the target. However, there are instances where the beam falls into the sea and is deflected further which makes it travel a longer distance than it would have if it was travelling a straight line without interruption.

### B-3.5 MULTIPLE ECHOES

Multiple echoes occur as a result of several reverberations of the echoes from a different ship's radar and from that of own ship multiple times. In this case, the display screen can show more than two or three targets detected [6]

## B-4 LIMITATIONS OF THE MARINE RADAR

### B-4.1 THE PROBLEM OF TARGET DISCRIMINATION

The need to distinguish among targets is crucial in any object detection application, especially in ship navigation. Distinguishing targets from one another will allow an operator to assess the level of threat posed by such targets and consequently, maneuvering decisions can be taken rapidly and conveniently. The two types of target discrimination are range and bearing discrimination. Range discrimination describes the ability of the radar system to display separately the echoes of two targets which lie on the same bearing but with slightly different ranges, while bearing discrimination is the ability of the radar set to distinguish between two different targets of the same range but with slightly different bearings.

### B-4.2 LACKING OF THE ABILITY TO CLASSIFY DETECTED TARGETS

Radar is good at detecting objects especially those far away from the receiver but it cannot classify them. This is due to the limitations of the technology of the microwaves / radio waves and the components used to display and analyze the radar data. The signals are not intelligent enough to tell the difference in object type especially when multiple objects are detected at the same time. Because of this, seafarers do not always get detailed information about the targets, to make the best maneuvering decisions.

### B-4.3 BLIND RANGE OF NO DETECTION

There is a minimum range below which an object is no more in the radar's scope. Also, large objects that are too close to the transmitter can saturate the receiver and thus, lead to a confused display of the detected object. The radio signals work best when the object is further away from the receiver and not closer. This is also, somehow, due to the height of the antenna and its location on the ship. So, this means that there is a dead zone (area in which no object can be detected through the radar) depending on the height of the antenna. Hence, the higher we have the antenna, the longer the range for object detection but the bigger dead spots around the ship.

### B-4.4 FALSE AND UNWANTED RADAR RESPONSES

Radar can be interfered with by several objects and mediums in the air. The radio signals face plenty of interference, from objects and particles like sea waves, land, fog, rain, birds, meteors and reflections from a ship. In addition to this, radio signals can also be combined with other radio signals from other frequencies and if not properly directed, the signals can be interrupted by other signals and alter the information being transmitted.[7]

# PART 2: UTILISATION OF ARTIFICIAL INTELLIGENCE AND CAMERAS TECHNOLOGIES FOR OBSTACLE DETECTION IN SHIP NAVIGATION

# SECTION A: BRIEF INTRODUCTION TO ARTIFICIAL INTELLIGENCE, MACHINE LEARNING AND DEEP LEARNING

## A – 1 GENERAL OVERVIEW OF AI

### A-1.1 AI DEFINITION

Before starting to explain the term AI, let us take you back to a very basic idea about what intelligence is. While intelligence is one of the most treated subjects in psychology, there is no standard definition of what exactly it constitutes. Some researchers have suggested that intelligence is a single, general ability. while others believe that it encompasses a range of aptitudes, skills, and talents. With such different interpretations, it becomes difficult to define biological or natural intelligence [14]. Likewise, it's even more difficult to define artificial intelligence in a simple and robust manner due to its inherent different interpretations. Because of this, there is also no single definition of AI within the scientific community. However, we will try using examples and historical definitions to characterize the field of AI. Through these definitions, one will understand in his own wisdom, what AI researchers and engineers are trying to solve in this field.

In 1955, John McCarthy, one of the pioneers of AI, was the first to define the term *artificial intelligence*, roughly as follows:

*"The goal of AI is to develop machines that behave as though they were intelligent"* [15].

Clearly, today's AI has the goal of solving difficult practical problems which are surely too demanding for the scope of this definition. In the Encyclopedia Britannica, one finds another definition that goes like this:

*"AI is the ability of a digital computer or computer-controlled robot to perform tasks that are normally associated with intelligent beings"* [16].

This definition also has a weakness. For example, it would admit that the memorization and fast computing ability of computers are demonstrations of intelligent capabilities, which are also certainly considered a high intellectual processing capability of humans. According to this definition, then, every computer is an AI system. This problem is solved elegantly by the following definition from Elaine Rich:

*"Artificial Intelligence is the study of how to make computers do things at which, at the moment, humans are better"* [15].

This definition concisely characterizes what AI researchers have been doing for the last 50 years and it will be up to date even in the year 2050.

The purpose of Artificial Intelligence is to aid human capabilities and help us make advanced decisions with far-reaching consequences. Researchers and developers in the field are making surprisingly rapid strides today in mimicking activities of humans such as learning, reasoning and perception and this is continuously evolving with the potential to benefit many different industries, like the maritime.

## A-1.2 AI HISTORY

The desire to mimic the human brain in some mathematical way became popular in the 1940s. A school of thought called "Connectionism" was developed to study the process of human cognition that utilizes mathematical models, known as connectionist networks or artificial neural networks. In 1943, two of the connectionism advocates, Warren S. McCulloch, a neuroscientist, and Walter Pitts, a logician, published their classic paper titled: "*A logical calculus of the ideas immanent in nervous activity*". In this paper, they try to understand how the brain could produce highly complex patterns by using many basic cells that are connected together [17][18].

In 1950, a man named Alan Turing published "Computing Machinery and Intelligence", proposing what is now known as the Turing Test, a method for determining if a machine is intelligent. He believed that if a machine could carry on a conversation by way of a teleprinter, imitating a human with no noticeable differences, the machine could be described as thinking. His publication was followed in 1952 by the Hodgkin-Huxley model of the brain as neurons forming an electrical network, with individual neurons firing in all-or-nothing (on/off) pulses. These events, at a conference sponsored by Dartmouth College in 1956, helped to spark the concept of Artificial Intelligence [19][20].

After starting as an exciting, imaginative concept in 1956, the development of AI has not been straight forward and efficient and research funding was cut in the late 1970s, after several reports criticized a lack of progress. Efforts to imitate the human brain, called "neural networks," were experimented with, and later dropped. The most impressive functional programs were only able to handle simplistic problems, and were described as toys by the unimpressed. AI researchers had to deal with two very basic limitations: not enough memory and processing speed that will seem absurd by today's standards. However, in the 1980s, Japan initiated a project called the "Fifth Generation Computer Systems (FGCS)" for the goal of becoming a world leader in computer technology. This triggers the upturn for funding in AI research in the US and UK to compete with the "fifth generation" computer project and it leads to the development of expert systems in the 1980s [19][20].

Between 1987 and 1993, AI research witnessed another cut in funding. This second slow-down in AI research was caused by the Expert Systems computers being seen as slow and clumsy. Also, this coincided with the surge of desktop computers, and the older, bulkier and much less user-friendly computers, were not the preferred choices. Expert Systems simply became too expensive to maintain, difficult to update and they could not 'learn'.

In the early 90s, the focus on AI research shifted to what we called '*Intelligent agents',* fed with natural language sample data collection. In 1997, a type of recurrent neural network called Long-Term-Short-Memory was developed for handwriting and speed recognition. Most of these algorithms were not very efficient due to low computing power (in comparison to today's standards) and the amount of data needed to run them at that time [19].

In the first decades of the 21$^{st}$ century, access to large amounts of data (known as "big data"); cheaper and faster computers and advanced artificial intelligence techniques were successfully applied to many problems throughout the economy. By 2016, advances in deep learning drove progress and research in image and video processing, text analyses and even speech recognition. The current availability of data, cheaper computing power and efficient algorithms has opened an era of implementation of some techniques of AI (machine learning) across almost all industries, and at the same time, opened new frontiers of research.

In Summary, the last decade has been immensely important in AI innovation. This is thanks to the massive developments in computer hardware and the availability of greater, faster and cheaper computing power, improvements in the AI algorithms and the availability of data (Big Data).

## A-1.3 CLASSIFICATION OF AI

General classification of AI into branches depends on many factors. It depends on whether we are classifying based on approaches (techniques), applications or fields of study. Some of the classifications below may be regarded as concepts or topics rather than full branches and some of them heavily overlap.

### *A-1.3.1 Stages of Artificial Intelligence*

Artificial intelligence generally falls under three broad stages: Artificial Narrow Intelligence, Artificial General Intelligence and Artificial Super Intelligence.

#### a. Artificial Narrow Intelligence (ANI)

Artificial narrow intelligence (ANI), also referred to as weak AI or narrow AI, is the only type of artificial intelligence we have successfully realized until today. Narrow AI is goal-oriented, meaning it is designed to perform singular tasks. Examples of such simple tasks include facial recognition, speech recognition/voice assistants, driving a car or searching the internet. Narrow AI is very good at performing the specific task it is programmed to do.


**Figure 7: Self-Driving Car [21]**

While these machines may seem intelligent, they operate under a narrow set of constraints and limitations, which is why this type is commonly referred to as weak AI. Narrow AI doesn't mimic or replicate human intelligence, it merely simulates human behavior based on a narrow range of parameters and contexts.

However, narrow AI has experienced numerous breakthroughs in the last decade, powered by achievements in machine learning and deep learning. For example, AI systems today are used in medicine to diagnose cancer and other diseases with extreme accuracy through replication of human-like cognition and reasoning.

#### b. Artificial General Intelligence (AGI)

AGI, sometimes referred to as Strong AI, is the stage in evolution of artificial intelligence wherein machines will possess the ability to think and make decisions just like humans. Strong AI uses a theory of mind AI framework, which refers to the ability to discern needs, emotions, beliefs and thought processes of other intelligent entities. Theory of mind level AI is not about replication or simulation, it's about training machines to truly understand humans. The immense challenge of achieving this stage of AI is not surprising when we consider that the human brain is the model for creating general intelligence. The lack of comprehensive knowledge on the functionality of the human brain has researchers struggling to replicate basic functions of sight and movement. There is currently no capability of AGI but researchers are still working to achieve it.

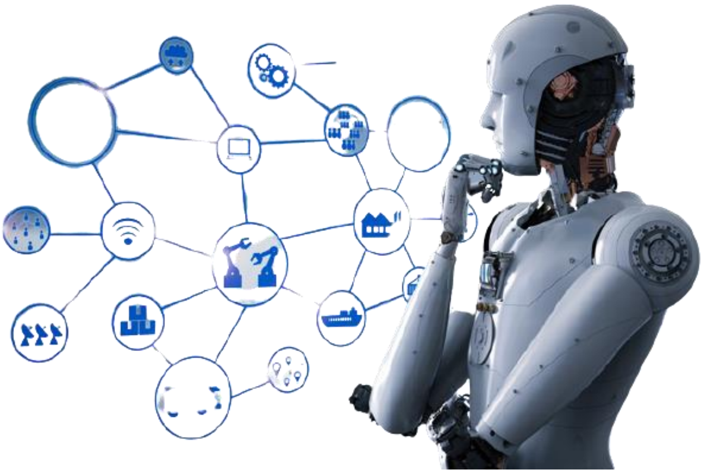


**Figure 8: AI at the stage of AGI [22]**

### c. Artificial Super Intelligence (ASI)

Artificial super intelligence (ASI), is the hypothetical AI that doesn't just mimic or understand human intelligence and behavior but becomes self-aware and surpasses the capacity of human intelligence and ability. This would include decision making, making better art and building emotional relationships. Once we achieve Artificial Super Intelligence, AI systems would rapidly be able to improve their capabilities and advance into domains that we might not even have dreamed of. The potential of having such powerful machines at our disposal may seem appealing, but the concept itself has a multitude of unknown consequences. If self-aware super intelligent beings came to be, they might be capable of ideas like self-preservation. The impact this will have on humanity, our survival, and our way of life, is pure speculation [23].

## *A-1.3.2 Branches of Artificial Intelligence*

### a. Machine Learning (ML)

ML is a science that enables machines and computer systems to process, analyze and interpret data with the aim of providing solutions for real-life challenges. Here are the three major categories under Machine Learning:

1. <u>Supervised Learning</u>: In this type of machine learning, the training data for the algorithm is labeled and variables are defined to the algorithm for accessing and finding correlations. Both the inputs and the outputs of the algorithm are defined.

2. Unsupervised Learning: This type of machine learning involves algorithms that are trained on unlabeled data. The algorithm analyzes the datasets to draw meaningful correlations such as hidden patterns in the dataset.

3. Reinforcement Learning: In this type of machine learning, the algorithm learns to achieve a goal in an uncertain, potentially complex environment. The algorithm faces a game-like situation where it employs trial and error to come up with a solution to the problem. To get the algorithm to do what the programmer wants, a positive or negative reward is given for the actions it performs. The goal of the algorithm is to maximize the total reward.

### b. Neural Networks (Deep Learning)

Neural Networks, also known as Artificial Neural Networks (ANN) or Simulated Neural Networks (SNN) are a subset of machine learning algorithms inspired by the human brain by mimicking the way that biological neurons signal to one another. A neuron in a neural network is a mathematical function whose work is to gather and classify information according to a particular structure and by implementing various statistical techniques to get a result. Deep Learning is the process of implementing Neural Networks on high dimensional data to gain insights and form solutions. Deep Learning is an advanced field of Machine Learning that can be used to solve more advanced problems.

### c. Natural Language Processing (NLP)

Natural language processing is a branch of artificial intelligence that helps computers understand, interpret and manipulate human language. NLP draws from many disciplines, including computer science and computational linguistics, in its pursuit to fill the gap between human communication and computer understanding.

### d. Robotics

Robotics is an interdisciplinary field of science and engineering incorporated with mechanical engineering, electrical engineering and computer science. AI robots have one or more artificial intelligence techniques embedded in them and they act in a real-world environment to produce results by taking accountable actions.

**Figure 9: Sophia, the humanoid robot, first unveiled to the world in 2016, is a good example of AI in robotics [24].**

### e. Fuzzy logic

The term Fuzzy means something that is a bit vague. When a situation is vague, the computer may not be able to produce a result that is True or False. As per Boolean Logic, the value 1 refers to True and 0 means False. But a Fuzzy Logic algorithm considers all the uncertainties of a problem, where there may be possible values besides True or False.

For example: **Boolean Logic** → Yes = 1 and No = 0

**Fuzzy Logic** → Very hot (0.9)

Little hot (0.20)

Moderately hot (0.35)

Not hot (1.0)

The term Fuzzy Logic was first described by Lotfi Zadeh in 1965. He thought that traditional computer logic is not capable of handling unclear or vague data. Similar to humans, there are many possible values between True and False that a computer can incorporate [25].

### f. Expert systems

An expert system is an AI-based computer system that learns and reciprocates the decision-making ability of a human expert. Expert systems are designed to solve complex problems by reasoning through sets of knowledge, represented primarily as rules rather than conventional code of procedure. In other words, it uses rule-base programming to derive conclusions [25].

## *A-1.3.3 Relationship Between AI, Machine Learning and Deep Learning*

Understanding the relationship between artificial intelligence, machine learning and deep learning is relatively easy. Let’s clarify it better as follows: Artificial Intelligence is a set of algorithms that tries to mimic human intelligence. Machine Learning is one of the approaches to building those AI algorithms, and deep learning is one of the machine learning techniques.

To be more explicit:

- Machine learning is a subset of AI, and it consists of the techniques that enable computers to figure things out from the data and deliver AI applications.
- Deep Learning, meanwhile, is a subset of machine learning that enables computers to solve more complex problems.

# A-2 DEEP LEARNING AND NEURAL NETWORKS

Most of the time, Deep Learning and Neural Networks are two terms that people use interchangeably, but there is some difference between the two.

## A-2.1 NEURAL NETWORK DEFINITION

Human brains are made up of connected networks of neurons. With Artificial Neural Networks (ANNs), researchers seek to simulate these networks and get computers to act like interconnected brain cells for them to be able to learn and make decisions in a more human-like manner.

Different parts of the human brain are responsible for processing different pieces of information, and these parts of the brain are claimed to be arranged hierarchically, or in layers. In this way, as information comes into the brain, each level of neurons (layer) processes the information, provides insight, and passes the information to the next, more senior layer.

This layered approach to processing information is what ANNs are trying to simulate. The simplest form of an ANN has only three layers of neurons, namely:

- <u>The input layer</u>: where the data enters the system
- <u>The hidden layer</u>: where the information is processed
- <u>The output layer</u>: where the system decides what to do based on the data

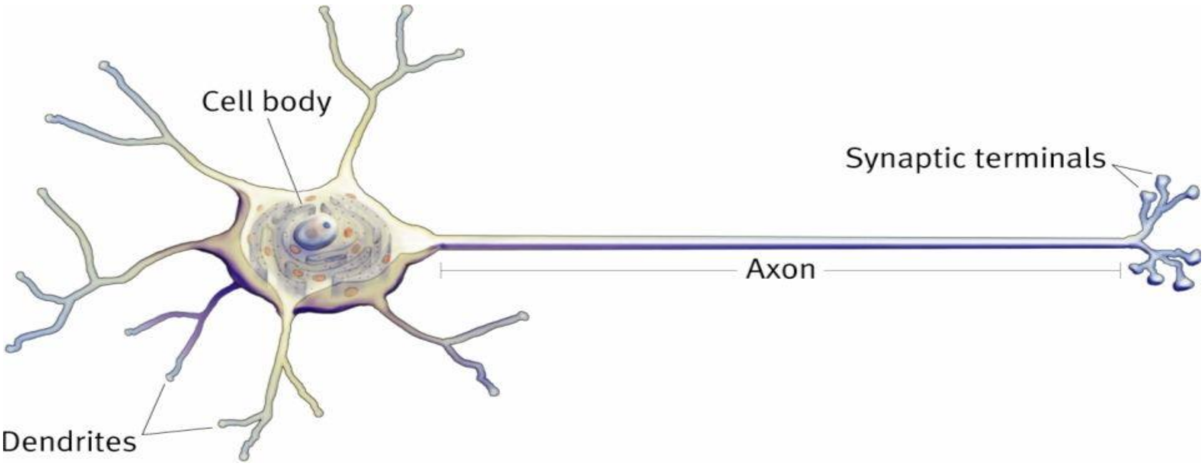


**Figure 10: A simple representation of a biological neuron [26].**

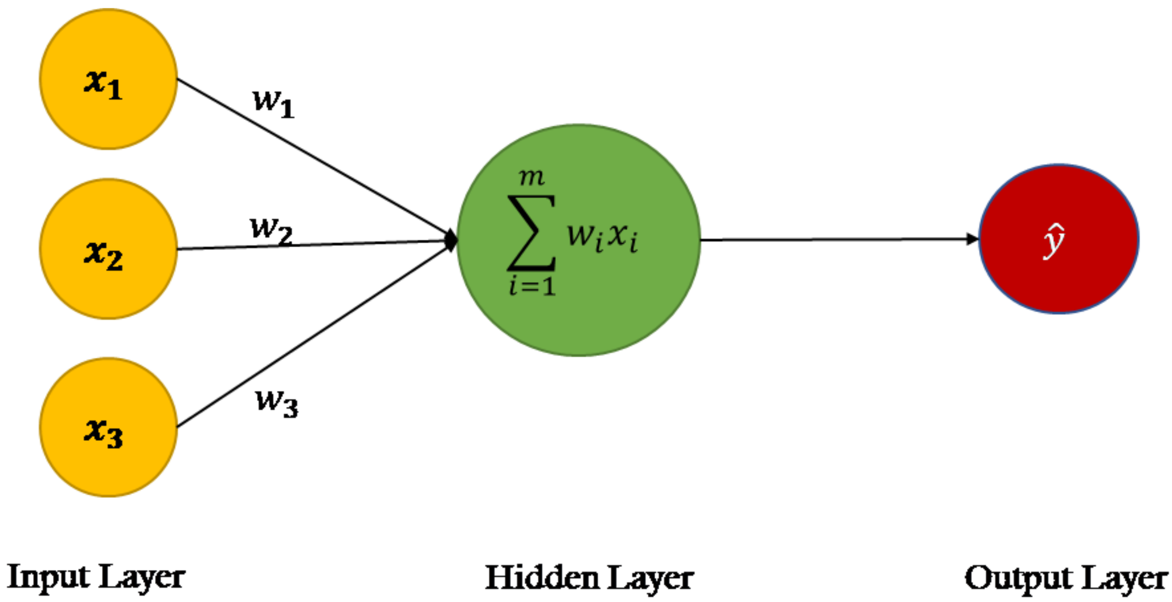



**Figure 11: representation of a simple artificial neural network: x1, x2, and x3 are the features of the input data; w1, w2 and w3 represent the weights: The weights are parameters that have to be trained using the input features and their labels and y is the output label [27].**

But ANNs can get much more complex than that, and include multiple hidden layers. Whether it's three layers or more, information flows from one layer to another, just like how some scientists supposed that the latter flows in the human brain.

## A-2.2 DEEP LEARNING DEFINITION

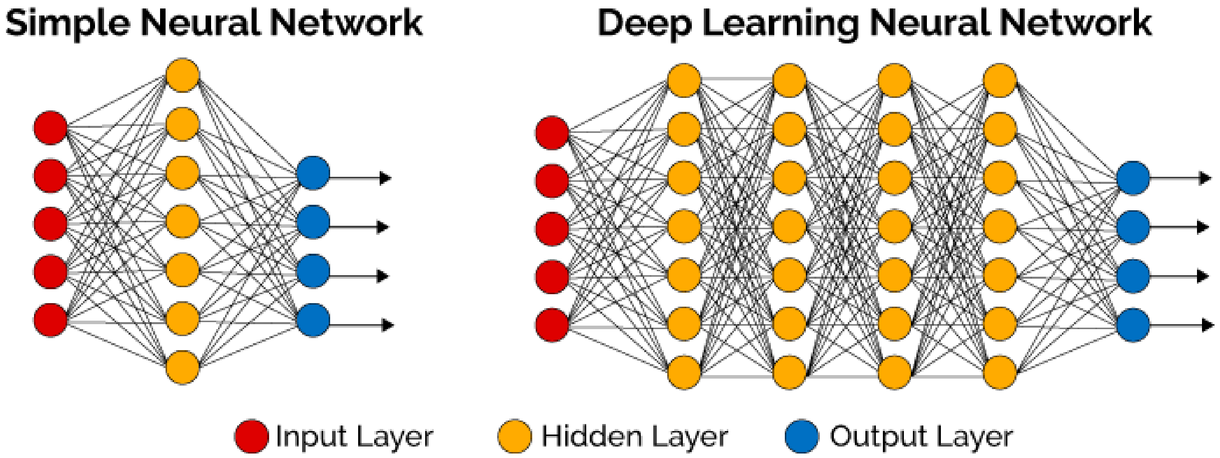



**Figure 12: Distinction between the representation of a simple neural network and deep learning [28].**

A neural network with multiple layers is called a deep neural network or more commonly a deep learning algorithm. Today, deep learning represents the very cutting-edge of AI (in supervised learning applications). If deep learning is a subset of machine learning, then how do they differ? Well, in the simplest terms, what sets deep learning apart from the rest of machine learning is how it processes the data and how it learns. Within each layer of the neural networks, the algorithms perform calculations and make predictions repeatedly, progressively learning and gradually improving the accuracy of the outcome over time.

The word 'deep' in deep learning is just referring to the depth of layers in artificial neural networks. As a result, ANNs that are made up of more than three layers i.e. an input layer, an output layer and multiple hidden layers can be considered as a "Deep Neural Network", and this is what underpins deep learning.

To sum up, we can say that these two terms are closely connected. That means that, without Neural Networks, there would be no deep learning. Hence, while Neural Networks use neurons to transmit data in the form of input values through connections, Deep learning is associated with the transformation and extraction of features which attempts to simulate a relationship between stimuli and associated neural responses in the brain.

# A-3 TYPES OF DEEP LEARNING ALGORITHMS

Different types of neural networks can be grouped into the following categories:

## A-3.1 MULTILAYER PERCEPTRON (MLP)

Also known as Feedforward Neural Networks, the Multilayer Perceptron are the first type of neural networks created and are the most common type used today. A perceptron is quite simply the most basic neural network that can be thought of while a multilayer perceptron has three or more layers (with the input layer included). In this network, every single node in a layer is connected to every node in the following layer and the information is sent through the multilayer perceptron in one direction without any feedback loops. These networks range from simple to complex depending on the number of hidden layers they contain.

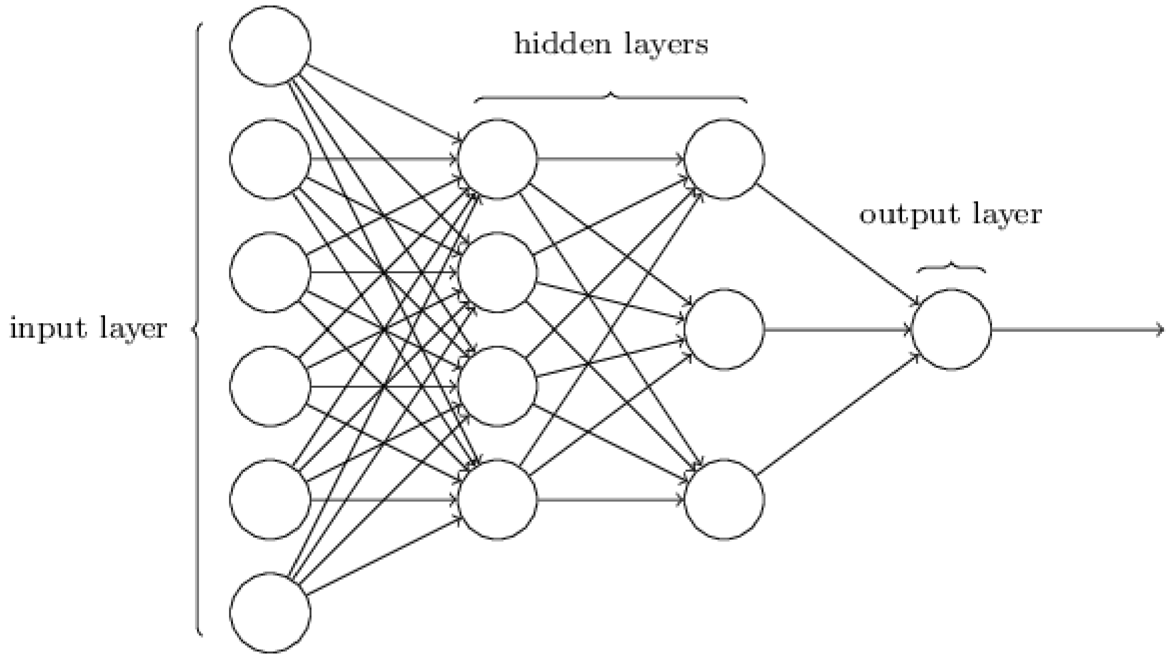



**Figure 13: Example of a multi-layer perceptron with two hidden layers, one input and one output layer [29].**

# A-3.2 CONVOLUTIONAL NEURAL NETWORKS (CNNs)

Convolutional Neural Networks, also Known as ConvNets, consist of multiple layers and are mainly used for image processing and object detection. Regarding its architecture, CNNs mainly contain convolutional layers, activation layers, pooling layers and Fully connected layers.

- Convolution Layer: CNN has a convolution layer that has several filters (also called kernels) to perform the convolution operations.
- Activation layer: perform operations on elements by computing them using a function called activation function. Examples of activation functions: ReLu, Softmax, Sigmoid, etc.
- Pooling Layer: The rectified feature map next feeds into a pooling layer. Pooling is a compression operation that reduces the dimensions of the feature map. The pooling layer then converts the resulting two-dimensional arrays from the pooled feature map into a single, long, continuous, linear vector by flattening it. Pooling layers are of two types: max pooling and average pooling.

➢ <u>Fully Connected Layer</u>: A fully connected layer forms when the flattened matrix from the pooling layer is fed as an input, which classifies and identifies the images.

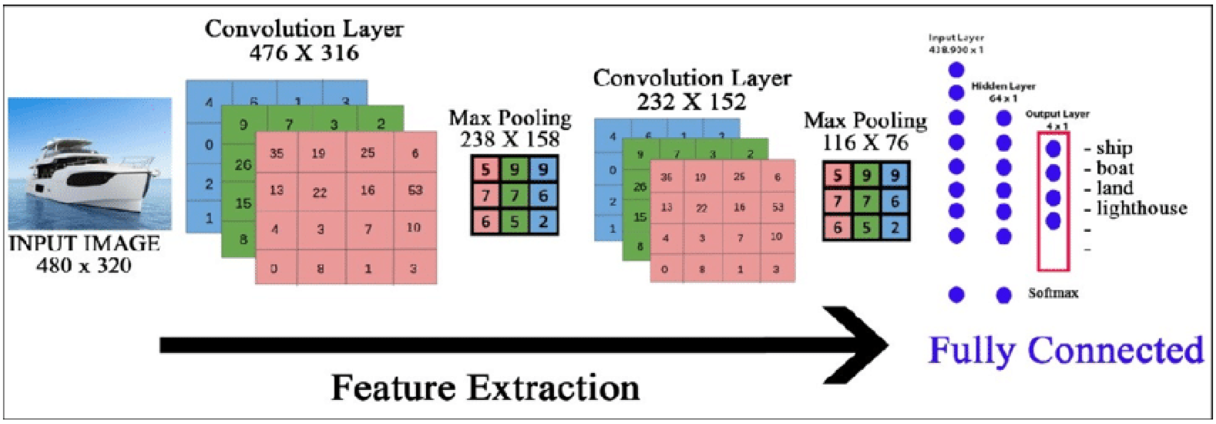



**Figure 14: Representation of the CNN architecture recognizing an image [30]**

# A-3.3 RECURRENT NEURAL NETWORKS (RNNs)

Recurrent neural networks allow information to flow in loops. As information is sent forward and backward, neurons retain data from the previous cycle. RNNs can perform more complicated tasks than Multilayer Perceptron, but they are slower. Long Short-term Memory Networks (LTMNs), a type of RNNs, contain memory cells that hold data from several previous cycles – longer than traditional RNNs. This makes RNNs ideal for sentence building in text and speech recognition [31].

# A-3.4 GENERATIVE ADVERSARIAL NETWORKS (GANs)

GANs are generative deep learning algorithms that create new data instances that resemble the training data. GAN has two components: a generator, which learns to generate fake data, and a discriminator, which learns from that false information. The usage of GANs has increased over a period of time. They can be used to improve astronomical images and simulate gravitational lensing for dark-matter research (an area of research in material science and nanotechnology)[31].

# A-3.5 MODULAR NEURAL NETWORKS (MNNs)

Modular Neural networks consist of two or more different types of neural networks working together to perform complex tasks. Each network operates independently on sub-tasks aimed toward the same output. MNNs are faster than having one network attempt the same task on its own and can also tackle exceptionally complex problems that individual networks cannot.

Research is underway using MNNs to develop biometric and human emotion recognition systems [31]

# SECTION B: OBJECT DETECTION

Detecting the presence of an object in an image and finding its location in it, is an object detection task. In other words, object detection includes both classification and localization of individual objects found in an image along with their corresponding bounding boxes. On the other hand, object tracking is defined as the problem of estimating the trajectory of a detected object in an image plane as it moves around a scene.

## B –1 DEFINITION OF OBJECT DETECTION, RECOGNITION AND TRACKING WITH DEEP LEARNING

### B-1.1 OBJECT DETECTION AND RECOGNITION

Image classification involves assigning a class label to an image, whereas object localization involves drawing a bounding box around one or more objects in an image. Object detection is more challenging as it combines these two tasks and draws a bounding box around each object of interest in the image and assigns them a class label. Together, all of these problems are referred to as object recognition. So, object recognition is a general term to describe a collection of related computer vision tasks that involve identifying objects in digital photographs.

- Image Classification: Classifying an image is when a computer analyzes an image and identifies the ‘class’ to which the image belongs to. A class is essentially a label, for instance, ‘car’, ‘animal’, ‘building’ and so on. For example: when we input an image of a ship, an image classification is the process by which the computer analyzes the image and tells that it’s a ship. (Or the probability that it’s a ship.)

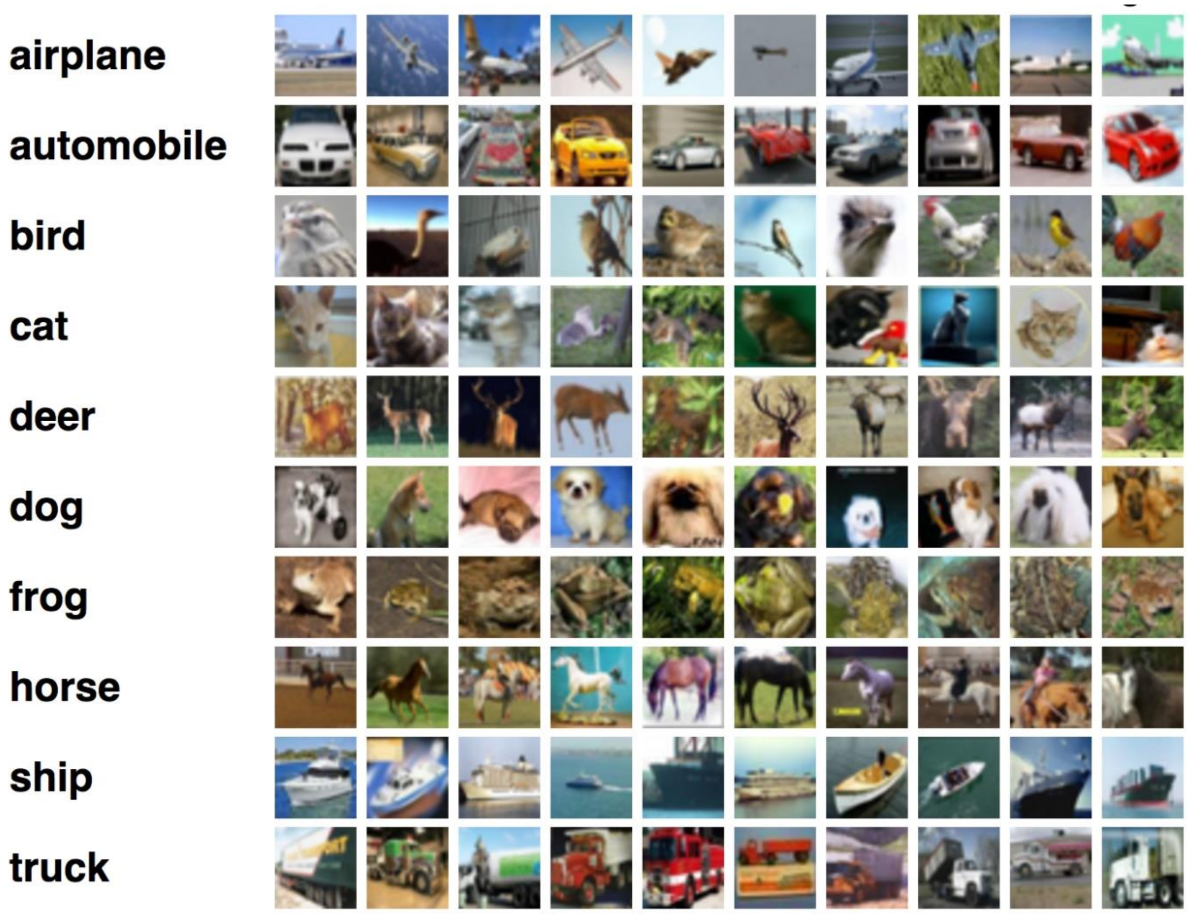



**Figure 15: Representation of some detectable classes[32].**

- Object Localization: It involves locating the presence of objects in an image and indicating their location with a bounding box.
    - Input: An image with one or more objects.
    - Output: One or more bounding boxes (defined by a point in the middle, a width, and a height)

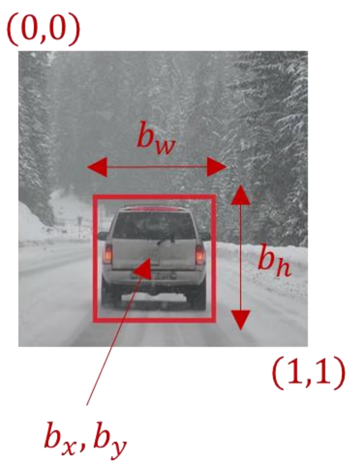



**Figure 16: technique of object localization (where the coordinates of the bounding box are defined) [33].**

**Here, the upper left point of the images is (0,0), lower right is (1,1). The bounding box is the red rectangle defined by the midpoint which is the point bx, by; as well as the height that would be bh, and bw the width of this bounding box.**

- Object Detection: It consists of locating the presence of objects with a bounding box, and identifying their types or classes in an image.
  - Input: An image with one or more object
  - Output: One or more bounding boxes (define by point, width, and height), and a class label for each bounding box

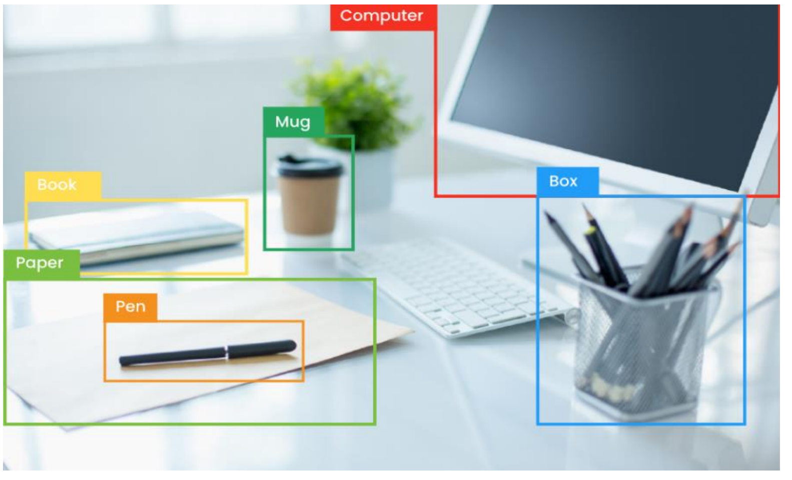



**Figure 17: Example of Object Detection [34].**

## B-1.2 OBJECT TRACKING

Object tracking is a discipline within computer vision, which aims to track objects as they move across a series of video frames. Objects are often people, but may also be animals, vehicles or other objects of interest such as ships in our case. Technically, object tracking starts with object detection identifying objects in an image and assigning them bounding boxes. The object tracking algorithm assigns an ID to each object identified in the image, and in the following images (frames) tries to carry across this ID and identify the new position of the same object [35].

We can distinguish mainly two levels of object tracking:

- Single Object Tracking: for this level of object tracking, the bounding box of the target (object) in the first frame is given to the tracker (to define the tracking algorithm). The goal of the tracker is then to locate the same target in all the other frames.

- Multiple Object Tracking: here there are multiple objects to track. The tracking algorithm is expected first to determine the number of objects in each frame, and second, to keep track of each object's identity from one frame to the next.

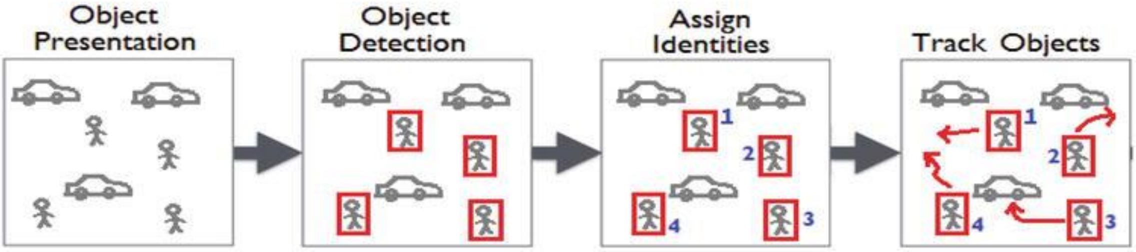



**Figure 18: Steps involved in multi-object tracking [36].**

# B–2 WORKING PRINCIPLE OF OBJECT DETECTION BASED ON DEEP LEARNING

In the early stages, before the deep learning era, the workflow of an object detection system was divided into three steps known as the traditional approaches. The problem with the traditional approach is that the tasks involved are computationally expensive and are not always sufficient to perfectly describe all types of objects. This takes us to the deep learning approach where the neural networks have achieved remarkable progress in object detection. They are able to learn more complex features as well as tackle many drawbacks found in the traditional approach.

The Deep Learning approach of object detection has two main components, an encoder, and a decoder. An encoder takes an image as input and runs it through a series of blocks and layers that learn to extract statistical features used to locate and label objects. Outputs from the encoder are then passed to a decoder, which predicts bounding boxes and labels for each object. The decoder uses the regression model to predict the location and size of the object.

There are two varieties of deep learning-based object detection algorithms, namely:

- Region proposal-based networks: Examples include R-CNN, Fast R-CNN, Faster R-CNN, cascade R-CNN, Mask R-CNN, SSP-net, etc.
- Regression-based networks: Examples include SSD, YOLO, etc.

## B-2.1 REGION PROPOSAL BASED

Region-CNN (R-CNN) is one of the state-of-the-art CNN-based deep learning object detection approaches. Its best varieties include fast R-CNN and Faster R-CNN for faster object detection. These algorithms are all based on the Region Proposal Network (RPN: is a fully convolutional network that simultaneously predicts object bounds and objectness scores at each position).

*(Objectness score: is defined to measure how well the detector identifies the locations and classes of objects during navigation.)*

## B-2.2 REGRESSION BASED ALGORITHMS

The regression-based algorithms use only CNN computations to map straightly from image pixels to bounding box coordinates and class probabilities; these reduce time and computational expense. It does not require any additional algorithm. Two of the state-of-the-art regression-based algorithms are detailed below

### *B-2.2.2 SSD Mobilenet*

- SSD – Single Shot Detector

  SSD was developed by google research team to center the need for models that can run real-time on embedded devices without a significant trade-off in accuracy. Single Shot object detection takes one single shot to detect multiple objects within the image. The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes. Its design architecture is composed of two parts:

  - Extraction of features
  - Application of Convolutional filters to detect objects

- MobileNet

  MobileNet is a light-weight deep neural network architecture designed for mobiles and embedded vision applications. It was developed to fulfill the requirements of many real-world applications such as self-driving cars, to carry out recognition tasks on computationally limited devices with an appropriate duration.

- SSD MobileNet

  To further tackle the practical limitations of running high resource and power-consuming neural networks on computationally limited devices in real-time applications, MobileNet was integrated into the SSD framework and hence result to the formation of SSD MobileNet.

  To summarize, MobileNet is used for classification and recognition whereas the SSD is a framework that is used to realize the multibox detector, and the combination of both performs object detection.

### *B-2.2.2 Yolo*

You Only Look Once (YOLO) is a state-of -the-art, real-time object detection system. It is also defined as a clever convolutional neural network (CNN) for doing object detection in real time. The algorithm applies a single neural network to the full image, and then divides the image into regions and predicts bounding boxes and probabilities for each region. These bounding boxes are weighted by the predicted probabilities. YOLO is popular because it achieves high accuracy while also being able to run in real-time. This algorithm has a number of advantages over other object detection methods:

- YOLO is extremely fast compared to other algorithms
- YOLO learns generalizable representations of objects so that when trained on natural images and tested on artwork, the algorithm outperforms other top detection methods.

However, the YOLO algorithm faced certain limits:

- YOLO has difficulty in dealing with small objects in groups, which is caused by strong spatial constraints imposed on bounding box predictions.
- YOLO also struggles to generalize to objects in new/unusual aspect ratios/configurations and produces relatively coarse features due to multiple compression operations [37].

# B–3 AN OVERVIEW OF VISUAL CAMERA SENSORS AND COMPUTING POWER REQUIREMENTS FOR OBJECT DETECTION WITH DEEP LEARNING

## B-3.1 AN OVERVIEW OF VISUAL CAMERA SENSORS FOR OBJECT DETECTION

There are six different types of cameras based on how they perceive the environment used for obstacle detection and distance measurement approaches: stereo, monocular, RGB-D, omnidirectional, fisheye and event-based cameras.

### *B-3.1.1 Stereo Camera*

A stereo camera is a camera that has two lenses about the same distance apart as our eyes and takes two pictures at the same time. This simulates the way we actually see and therefore creates the 3D effect when viewed. The position of objects in the 3D image can be accurately obtained by extracting and tracking key information between two pairs of stereo images and then applying a motion estimation algorithm.

### *B-3.1.2 Monocular Camera*

A monocular camera is one which houses only one lens rather than two. It essentially functions like a small telescope by bringing faraway objects into sharp near focus. They are a very common type of vision sensor used in automated-driving applications.

### *B-3.1.3 RGB-D Camera*
They are a specific type of depth sensing devices that work in association with a normal camera, that are able to augment the conventional image with depth information (related with the distance to the sensor) in a per-pixel basis.[43]

### *B-3.1.4 Omnidirectional Camera*
An omnidirectional camera, also known as 360-camera, is a camera with 360 degrees field of view (fov) in azimuth and 90 to 140 degrees in elevation. It can achieve more accurate pose estimation compared with traditional cameras with a small field of view, because it can capture more information from the environment.

### *B-3.1.5 Fisheye Camera*
This is a type of camera that uses an extremely wide-angle lens to capture a viewing range of about 180 degrees. This type of camera is called a “fish eye” because it approximates the vision of a fish’s convex eye. The lens produces curvilinear images of a large area, while distorting the perspective and angles of objects in the image.

### *B-3.1.6 Event-Based Camera*
An event camera, also known as a neuromorphic camera, silicon retina or dynamic vision sensor is a bio-inspired imaging sensor that responds to local changes in brightness [38].

## B–3.2 COMPUTING POWER REQUIREMENTS FOR OBJECT DETECTION WITH DEEP LEARNING
Deep learning requires a tremendous amount of computing power. High performance graphical processing units (GPUs) are ideal because they can handle a large volume of calculations in multiple cores with large memory available. However, it really depends on the model used for object detection.

# SECTION C: PROPOSITION OF A REAL SOLUTION AND ITS ADVANTAGES ON SHIP NAVIGATION SAFETY

## C-1 PRESENTATION OF THE PROPOSED ARCHITECTURE OF THE REAL SOLUTION

The smart obstacle detection system can improve safety navigation thanks to obstacle detection with deep learning and/or sensor fusion. Generally, two possible solutions can be realized. One with cameras as the only sensory inputs and another which combines both cameras and some traditional onboard equipment as inputs.

### C.1.1 ARCHITECTURE WITH CAMERA-ONLY AS EXTERNAL SENSORS

Today cameras have become a common and useful remote sensing instrument for vessels. The system will consist of multiple external cameras as sensors taking real-time images from the environment. A means of communication then transfers the data to the fast onboard computer processors (GPUs, FPGAs, …) where an intelligent algorithm processes the real-time data before displaying the results using a monitor.

*(FPGA: Field Programmable Gate Array)*

Each component of the overall architecture is detailed below:

#### *C-1.1.1 Combination of Long and Short-Range Cameras*

Multiple cameras will be feasible for better vision – some on the front, aft, starboard and port side of the ship. These onboard cameras are placed in such a way that they take pictures of the real-world environment and provide 360 degrees view all together. The cameras are connected to the onboard computer processors where the software processes the real-time data using deep learning algorithms.

### C-1.1.2 Wired or Wireless Communication

When using closed-circuit cameras, high quality cables are required since the data has to be displayed in real-time. In this case, internet connectivity might not be necessary for real-time data transfer and inference. On the other hand, IP cameras can be used with the availability of a high bandwidth local area network for the transfer of the real-time data to the processors. This does not seem to be a very robust solution due to numerous disadvantages when data is transferred within a wireless network. Therefore, using closed-circuit (wired) cameras seems to be the most efficient solution since there will be no problem of low bandwidth and the risk of cyber security attacks will be highly reduced.

### C-1.1.3 Onboard Computer Processors

Having the right infrastructure in terms of hardware computing, mostly GPUs (Graphics Processing Units), is crucial as it increases the system throughput and drastically reduces the risk of failure. The ability of various deep learning algorithms to utilize the GPUs to speed up computations is also a huge benefit by reducing system latency. The more powerful the GPUs, the faster the detection and tracking process and the more we tend towards a real-time analysis of the environment.

### C-1.1.4 AI Software

The AI software is the most crucial element of the system. It’s composed of deep learning algorithms which perform inference on the real-time data from the cameras by classifying, outlining a bounding box and tracking the detected objects. The algorithmic model has to be trained with a large amount of maritime traffic data, in cloudy and foggy as well as rough conditions or in any uncommon weather to make them robust when deployed. They also have to make sure that buoys and debris are classified as separate categories, as well as ensuring flying objects are classified as false alarms (in case they fly within the camera’s range). In addition, the model should be trained and fine-tuned in such a way that it becomes resilient against new boat models and regularly updated for consistent performance. The algorithms will be able to determine the characteristics of the detected objects such as its class (what object is it), speed, bearing, distance and threat level of collision. The object will be classified based on what it is, whether it is a tanker, cargo ship, buoy, zodiac, fishing boat, offshore windmill etc.

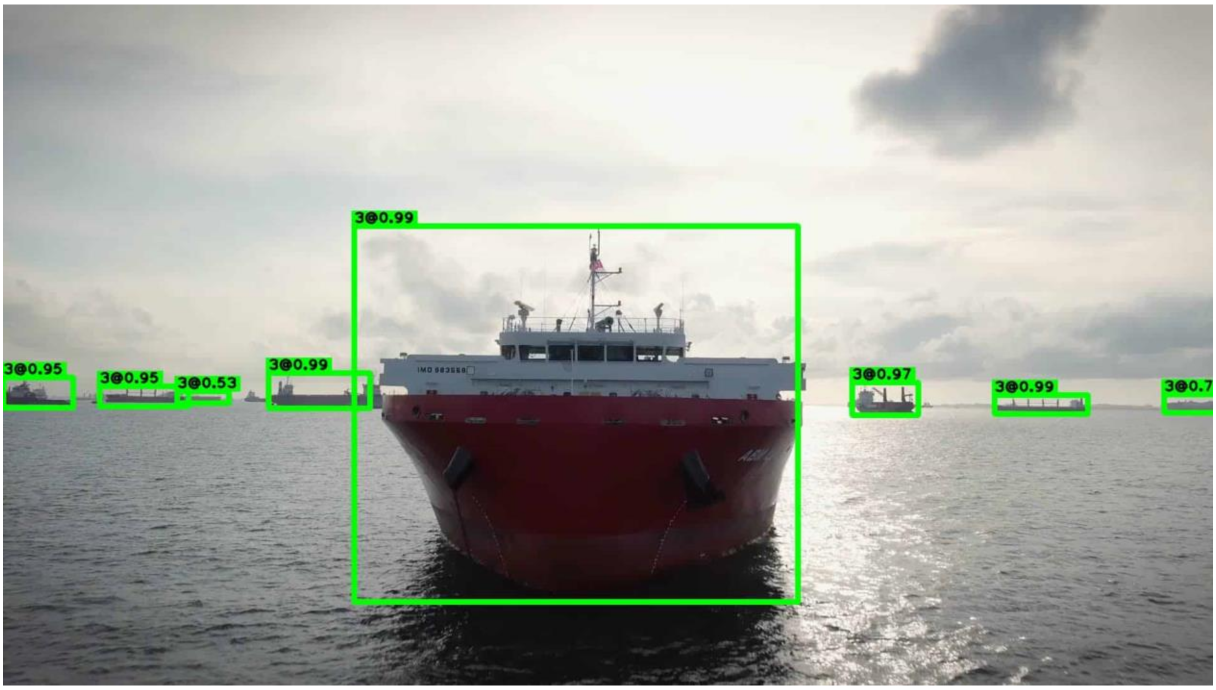



**Figure 19: Example of ships detected with AI software (using deep learning algorithm) [39].**

### *C-1.1.5 Monitor*

This is the device through which the processed data will be displayed. The environment will be monitored automatically and risk evaluation information will also be displayed when necessary. When the risk level exceeds a certain threshold, a series of alarms will be triggered each corresponding to a particular threat level. The operator will be more aware of the magnitude of the danger and therefore will take appropriate steps to neutralize it. Ships today require constant human monitoring and when the problem of fatigue and stress arrives, accidents become more prone. The limitations in the ability of humans to constantly survey and monitor the environment using only traditional technologies is therefore the main motivation for building the smart object detection system.


**Figure 20: A typical image of the environment that can be displayed by a monitor [39].**

## C-1.2 SENSOR FUSION ARCHITECTURE

Although we presented a camera-only solution in the first architecture, the combined system – radar sensing, AIS (Automatic Identification System) and object detection with vision cameras will make the entire system more accurate. This involves the use of sensory data from the various sources mentioned above, combined into one comprehensive result. The process is called sensor fusion and it uses the combined advantage of all the sensors involved to provide a richer and more exact output information. Below is a list of various devices that are crucial by being an integral part of the sensor fusion architecture.

### *C-1.2.1 The Radar*

Radar is going to be useful for long range detection of objects or targets for the system. It is one of the best sensors to be fused with a camera. Regarding distance, speed and bearing of objects, a radar's measurement of such characteristics is more accurate than that of a camera. So, making the radar an integral part of the sensor fusion architecture will ensure greater accuracy.

### *C-1.2.2 AIS – Automatic Identification System*

It provides a wealth of information on nearby vessels related to position, speed, identity, and routing. The AIS uses VHF radio frequencies to wirelessly broadcast the ships location, destination and identity to nearby receiver devices on other ships and land-based systems. AIS is very effective at monitoring ships which are legally required to install a VHF transponder, but will fail to detect those which are not, and those which disconnect their transponders [40].

### C-1.2.3 ECDIS – Electronic Chart Display Information System

It displays electronic navigation chart information that should represent the most recent hydrographic surveys of the areas sailed, the locations of channels and aids to navigation, and known hazards to navigation likely to be encountered along the route. [41]

### C-1.2.4 GNSS – Global Navigation Satellite System

It is a source for all of the above information in terms of vessel traffic position, speed and direction of transit.

The equipment listed above are all required for conventional merchant ships to install and they perform very well in extending the sight of the seafarers at sea to accomplish navigation functions. However, the IMO regulatory framework does not keep up with new technologies that can also enhance the safety of navigation for conventional ships (Using the word “conventional” to distinguish autonomous ships). These new technologies combined with existing traditional ones can take ship safety, security and efficiency to a next level. The fusion of existing technologies with new ones is a powerful way to overcome the current limitations of navigational vision. Each sensor type has inherent strengths and weaknesses. Radars are very strong at accurately determining distance and speed of targets – even in challenging weather conditions – but can’t classify or recognize the features of the targets. Cameras along with deep learning do very well at determining the classification, angular position and context of the scene of targets but can easily be blinded by dirt or sun. AIS is also very effective at monitoring ships (those with connected VHF transponders) by also providing position, speed, identity and routing. In case of target ships without VHF transponders, only data from the radar and camera will contribute to the fusion. There is also the possibility of adding the ECDIS and GNSS data for a more complex and detailed model of the environment. In this architecture, a combination of the camera, radar and AIS suffices for a moderate sensor fusion solution.

An example of a deep learning sensor fusion architecture is the Camera Radar Fusion-Net (CRF-Net) architecture [42] which was specifically built to fuse camera and radar sensor data of road vehicles. The network can be tweaked for offshore obstacle detection. A method called transfer learning can also be used where a model developed for a particular task is used as a building block to solve a different problem. In this case, the model has been trained and has the ability to detect many low-level features. Therefore, in transfer learning, little training and a relatively small dataset is required. The CRF-Net shows that the fusion network is able to outperform state-of-the-art image-only networks for two different datasets.

# C-2 ADVANTAGES OF THE SOLUTION IN SHIP NAVIGATION SAFETY

- The smart obstacle detection will provide more productive, predictable and safer marine operations.
- It is going to reduce incidents related to poor visibility and fatigue.
- Enable the crew to avoid collision by alerting them.
- It has the ability to detect objects within the radar’s blind zone (which is also very useful during docking).
- It is going to detect and classify all the vessels irrespective of being registered with IMO or with disconnected transponders.
- It solves the problem of radar discrimination by being able to distinguish several objects even if they lie together within the same bandwidth of the radar.
- It provides a more comprehensive and intuitive view of the environment ready to be analyzed by the user.
- There is no problem with wave interference since no wave is emitted.
- This system might try to avoid all of radar errors.
- It performs in multi-object tracking and situational awareness of the sea surface, in real time.

# PART 3: PROTOTYPING OF THE SMART OBSTACLE DETECTION SYSTEM

# SECTION A: SYSTEM'S ANALYSIS & FEASIBILITY STUDIES

## A-1 SYSTEM'S ANALYSIS

The aim of the project is to propose a smart obstacle detection for ship navigation assistance that is based on deep learning (as the Artificial Intelligence software) and camera sensor fusion. The system should be able to detect obstacles and analyze the environment in real time and give warnings regarding collision avoidance, thus helping the crew to make the necessary maneuvers. Below is a detailed overview defining the system's requirements and the functional analysis. This is a necessity to better refine the aim and mission of the project.

### A-1.1 SYSTEM'S REQUIREMENTS OF THE PROJECT

The idea of this project is to build a system that is able to detect fixed obstacles as well as mobile ones, in both day and night time. Images captured in real-time by cameras attached to the ship's exterior hull are passed through the deep learning algorithm to promptly report obstacles that could put the ship at risk. Going to this, the requirements for the project can be divided as follows:

#### *A-1.1.1 Hardware Requirements*

Below is a list of hardware components required for the project:

- Raspberry-pi board
- External USB webcam
- USB SD-card adapter
- Ultrasonic sensor
- Breadboard
- Jumper wires
- Computer for writing and training the algorithm and displaying the processed data
- LEDs & resistors

#### *A-1.1.2 Software Requirements*

Mainly, the software requirements will consist of an Artificial Intelligence algorithm (Deep Learning) and a programming language using Python and its corresponding frameworks.

##### a) Deep Learning Algorithm

The type of deep learning algorithm that is more appropriate for object detection is the Convolutional Neural Network (CNN). However, there are many algorithms based on CNN architecture that can be chosen for this project. Regarding the evaluation criteria of these algorithms, the SSD-MOBILENET has shown great performance in both speed and accuracy on real-time object detection over others. As a result, it is our choice of deep learning algorithm. Knowing that we are going to implement our algorithm in an embedded system like Raspberry Pi, a variation of SSD-MOBILENET called SSDlite-MOBILENET is going to be used specially for this project.

### b) Python language, Machine Learning libraries and development environment

- Python is favored by developers for a whole host of applications. It is renowned for its concise, readable code, and is almost unrivaled when it comes to ease of use and simplicity, particularly for beginners in programming.
  But what makes it a particularly good fit for projects involving AI like ours?
  One of the aspects is its abundance of libraries and frameworks that facilitate coding and save development time and for which deep learning is exceptionally well suited for. Python is also the language used for programming the raspberry pi hence its usefulness for this project.

- The best libraries to implement deep learning for real time object detection are OpenCV (Open Source Computer Vision) and TensorFlow, and both of them are going to be used for this project. The former (OpenCV) is a library of programming functions mainly aimed at real-time image processing and computer vision while the latter (TF) is a symbolic math library used for machine learning applications such as neural networks.

- GOOGLE COLABORATORY is an online platform that allows us to write and execute python code by providing free access to GPUs (Graphic Processing Unit). This platform will be used to train the deep learning algorithm.

## *A-1.1.3 Project Management*

The workflow of this project is divided into steps:

- Development of the algorithm.
- Implementation of the algorithm developed in the Raspberry Pi.
- Connection of the electronic components.
- Configuration and execution of the ultrasonic sensor program along with the LEDs.
- Emulation of the prototype.

# A-1.2 FUNCTIONAL ANALYSIS

In this part we are going to define and characterize the entire structure of the Obstacle Detection System for a better comprehension and a simpler overview of the project.

## *A-1.2.1 Functional Analysis of Requirements*

Analyzing the needs of the project will take us to establish the following diagrams:

a) Bull Chart

Bull Chart is one of the tools of functional analysis. It's a diagram that allows us to simply identify the major design objectives of a project. This diagram will also allow us to respond to three questions: to whom is this service useful? On what is it going to act? and what is its goal?

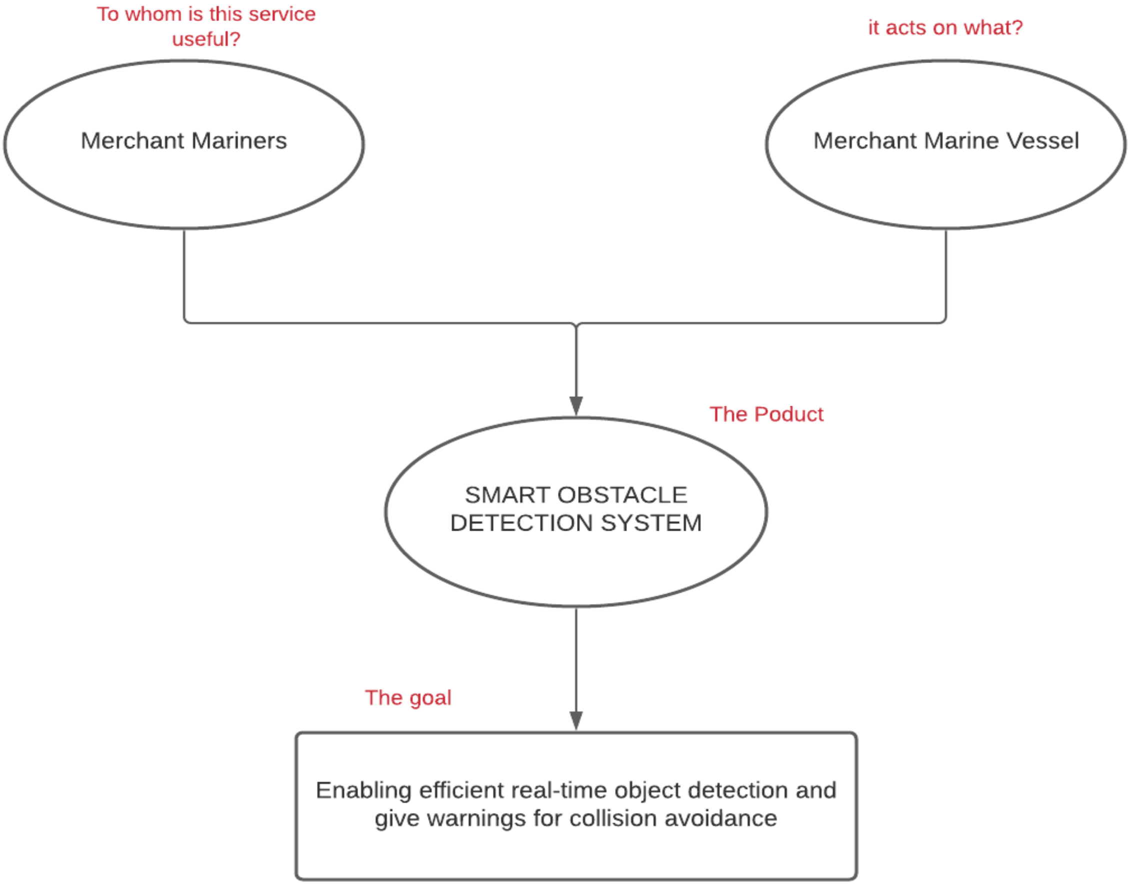



**Figure 21: Bull Chart [authors]**

b) Octopus Diagram

This diagram presents the different elements of the environment and the system to be designed. It allows the listing of the different service functions of the system, as well as the constraint functions.

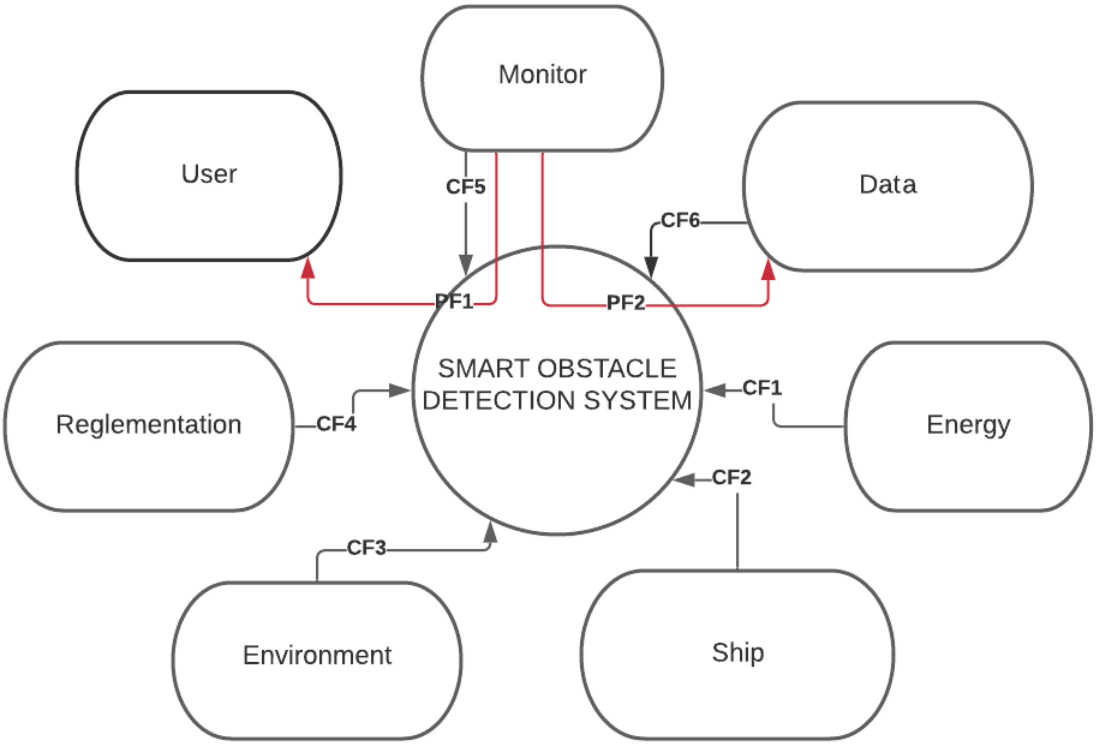



**Figure 22: Octopus Diagram [authors]**

In the following table, we are going to define the Principal Functions (PF) and the Constraint Functions (CF) of the system.

| | |
|---|---|
| PF1 | To notify the user to avoid collision by displaying real-time evaluated view of the environment and generating signs to indicate threat level |
| PF2 | To enable classification and detection of the obstacles |
| CF1 | To supply the components of the system with electricity. |
| CF2 | To be able to be implemented in every merchant ship system |
| CF3 | To provide real-time object detection coming from the marine environment, with good accuracy |

| | |
|---|---|
| CF4 | Related to the COLREG (International Regulations for preventing collisions at sea) regarding its goal of avoiding collision |
| CF5 | To display the images coming from the camera which is already processed by the deep learning algorithm |
| CF6 | Dataset used to train the deep learning algorithm |

## *A-1.2.2 Technical Functional Analysis*

a) Functional Analysis System Technique (FAST)

The Functional Analysis System Technique aids in thinking about the problem objectively and in identifying the scope of the project by showing the logical relationships between functions.

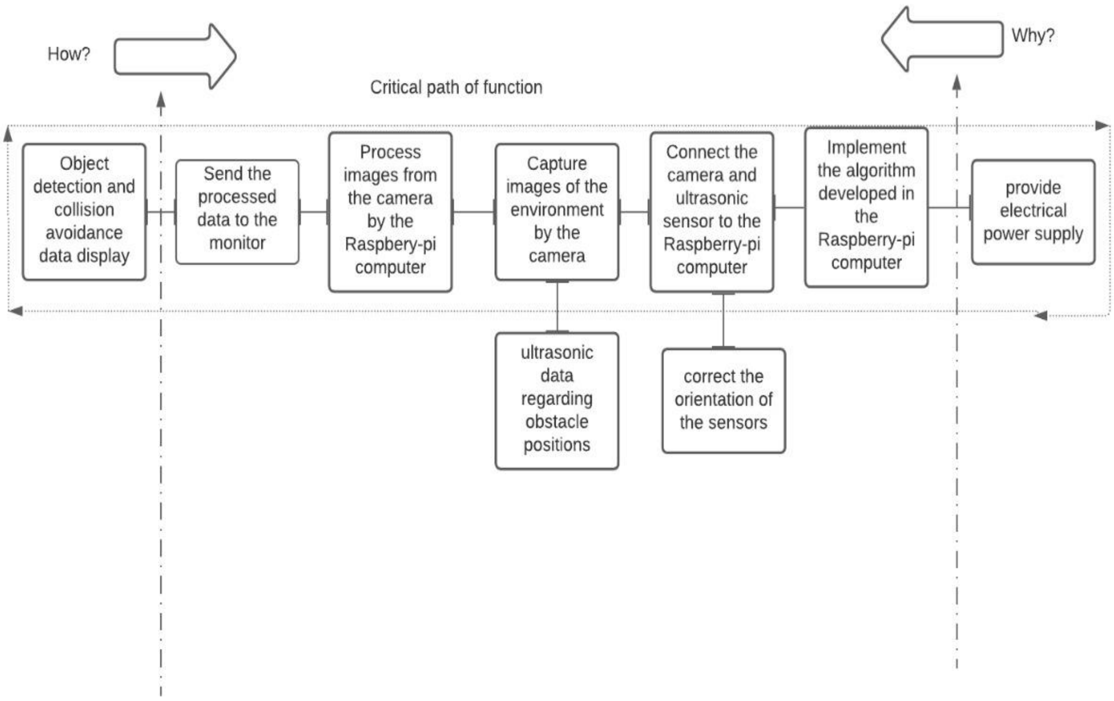



**Figure 23: FAST diagram [authors]**

b) Structured Analysis and Design Technique (SADT)

SADT is a diagrammatic notation designed specifically to help people describe and understand systems. It consists of box-arrow diagrams (blocks), with four arrows on each side defined as: input, output, control, mechanism and one activity.

Their definitions consist of the following:

- Activity: any function or process that serves to transform inputs into outputs.
- Input: the data/information required by an activity to start the transformation process.
- Output: the data/information produced by the activity as a result of this transformation.
- Control: any constraint that affects the behavior of activity in some way.
- Mechanism: persons, resources, or any means that are required to run the activity.

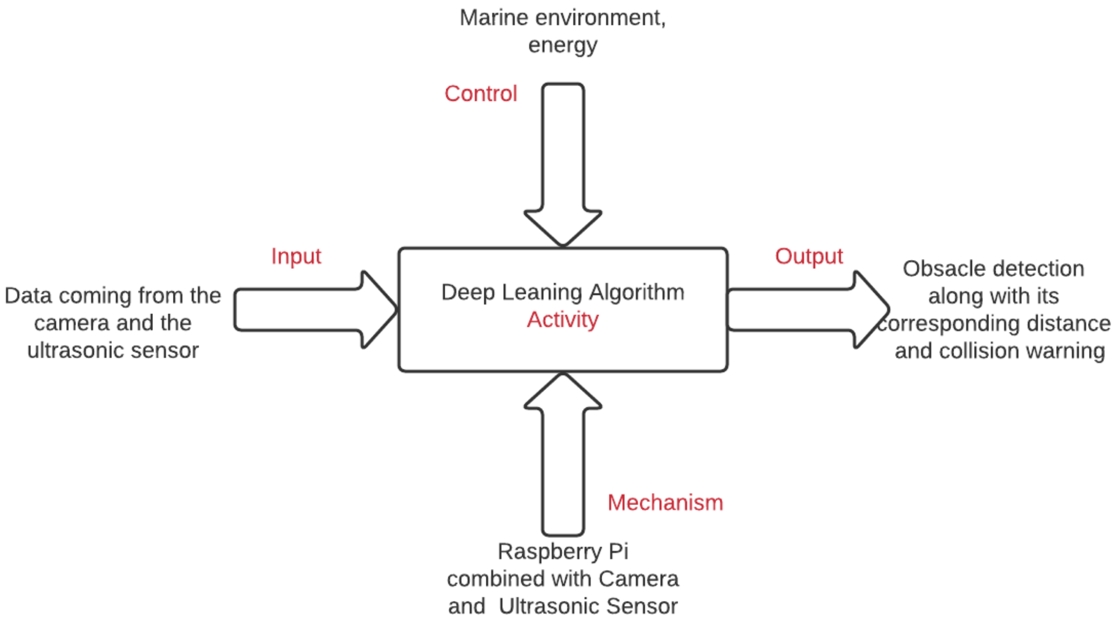



**Figure 24: First level SADT diagram [authors]**

# A-2 FEASIBILITY STUDIES

## A-2.1 LEGAL FEASIBILITY

The sea navigation is mainly regulated by the COLREGs (International Regulations for Preventing Collisions at Sea) to avoid ship collision. This convention also treats the requirement of certain equipment acting in the conduct of the ship. The obstacle detection system proposed in this project provides only object detection and collision avoidance warning but does not give maneuvering decisions. Although, equipment like the ARPA which give maneuvering decisions are directly treated by COLREGs, the observational equipment too such as the RADAR, or AIS must fall into the minimum COLREGs requirements.

The system proposed, which can also be considered an equivalent observational equipment as the radar, must therefore satisfy the minimum requirements of the COLREGs. In accordance with Rule 7 (b) (Risk of collision), "*proper use shall be made of radar equipment if fitted and operational, including long-range scanning to obtain early warning of risk of collision and radar plotting or equivalent systematic observation of detected objects*." Since our system with camera only (combined with ultrasonic sensor) has a very limited range for object detection, we introduce the camera-Radar sensor fusion architecture to satisfy the long-range scanning. This fusion, although not an integral part of the project, is thought as if the obstacle detection system is implemented in a real merchant ship.

## A-2.2 TECHNICAL FEASIBILITY

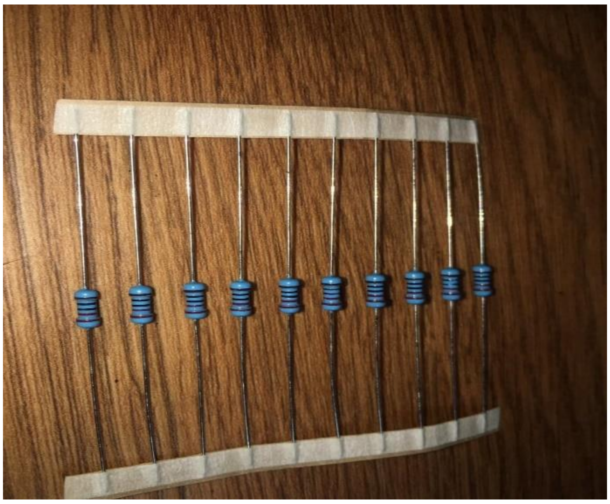
The realization of the smart obstacle detection system will inevitably come with many technical challenges like requiring:

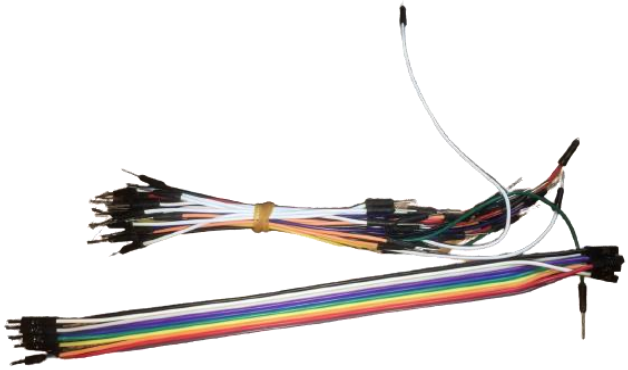
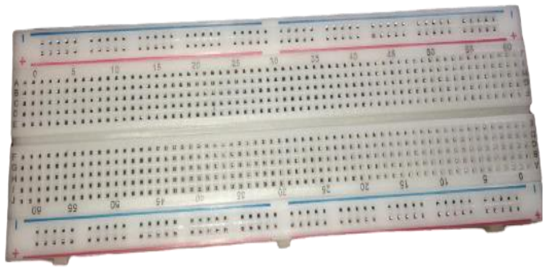
- Generation of a large amount of valuable dataset
- Good visibility with the camera
- Enough computing power to enable the embedded system to process the deep learning algorithm
- To resolve problems that can be faced by the camera due to the weather such as fog, rain, sun light and also due to the environmental hazards such as dust, corrosion etc.
- Enough maritime traffic data to train the algorithm to be very effective
- Enough data storage in the embedded system (Raspberry Pi) as it contains a limited memory storage capacity.
- Detection of obstacles with an acceptable accuracy

## A-2.3 ECONOMICAL FEASIBILITY

| COMPONENT'S NAME | REPRESENTATION | ROLE | PRICE |
| --- | --- | --- | --- |
| SD CARD (16 GB) |  | It is used as a storage device and to house the operating system of the raspberry pi. | 50DH |

| | | | |
|---|---|---|---|
| USB SD-CARD ADAPTER |  | It is used to allow the computer to access the SD-CARD and thus install the necessary software packages for the raspberry pi. | 25DH |
| RASPBERRY PI BOARD |  | It is the main component of the prototype. It acts as the brain of the system by providing an interface for the other components and providing computing power to process data. | 600DH |
| USB-CAMERA |  | It is used as an external sensor that takes real-time data from the environment to be processed by the deep-learning algorithm | 100DH |

| | | | |
|---|---|---|---|
| ULTRASONIC SENSOR |  | It acts as an external sensor, to calculate the distance of the detected objects with the help of the raspberry pi. | 120DH |
| LEDS (GREEN-YELLOW-RED) |  | The LEDs are used to give the warnings depending on the proximity of the obstacle. | 10DH |
| RESISTORS (18 resistors of 220 Ohm) |  | The resistors are essential in the circuit of the prototype to prevent power surge of the components. | 10DH |

| | | | |
|---|---|---|---|
| JUMPER WIRES |  | They are used to connect the components of the system between them. | 25DH |
| BREADBOARD |  | It is used to build and support the circuit of the system. | 87DH |
| MICRO-USB CHARGER |  | It is used to power up the raspberry pi. | 30DH |
| TOTAL | | | 1057DH |

# SECTION B: DESCRIPTION OF THE PROPOSED SYSTEM ARCHITECTURE

## B-1 DESCRIPTION OF THE COMPONENTS OF THE SYSTEM

### B-1.1 DESCRIPTION OF THE RASPBERRY PI BOARD

The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a cross between a typical embedded system – like an Arduino – and a desktop computer. It packs the different ports used to control it and also to control other devices:

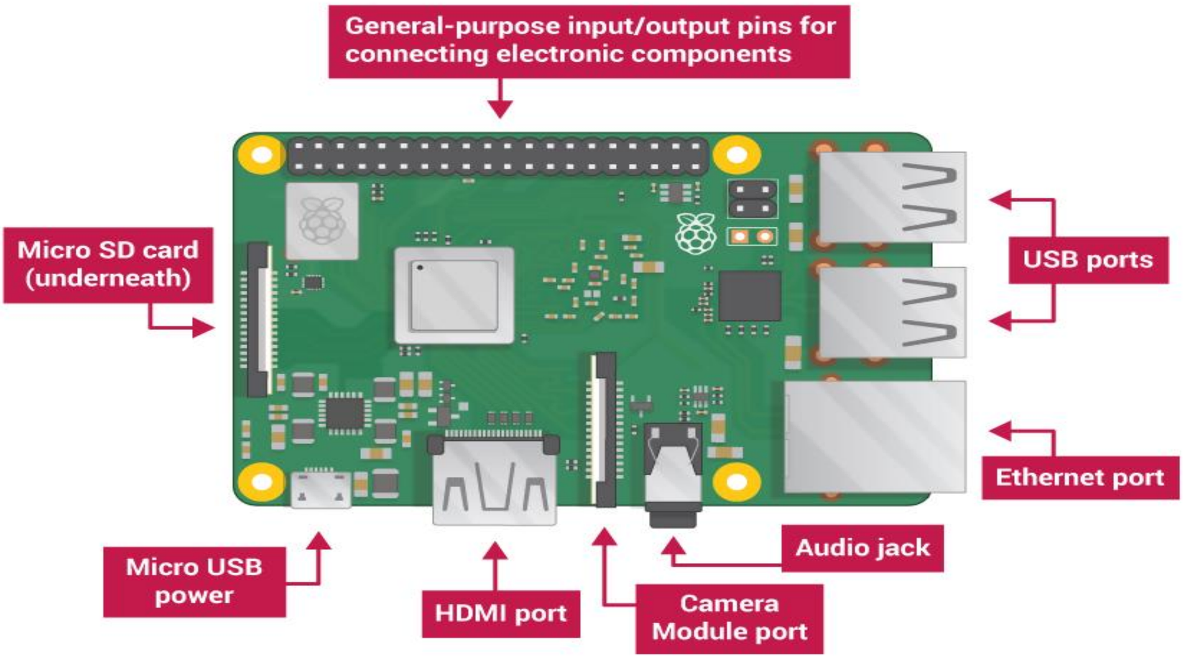
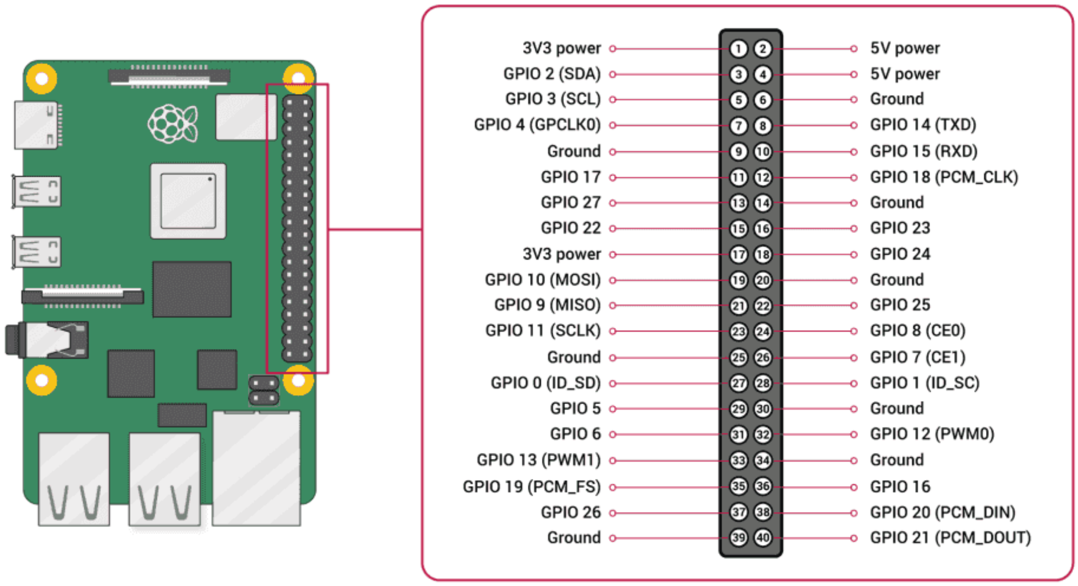



**Figure 25: Raspberry pi board**

- USB – USB ports are used to connect a wide variety of components, most commonly a mouse and keyboard.
- HDMI – The HDMI port outputs video and audio to the monitor.
- Audio – The audio jack allows you to connect standard headphones and speakers.
- Micro USB power – The Micro USB port is only for power. It should be powered only after everything else has already been connected.
- GPIO – The GPIO ports allow the Raspberry Pi to control and take input from any electronic component.




- SD card slot – The Raspberry Pi uses SD cards the same way a full-size computer uses a hard drive. The SD card provides the Raspberry Pi with internal memory, and stores the operating system.

There are different versions of the raspberry pi and one of them is the raspberry pi 3. Specifically, the raspberry pi 3 model B is used for this project.

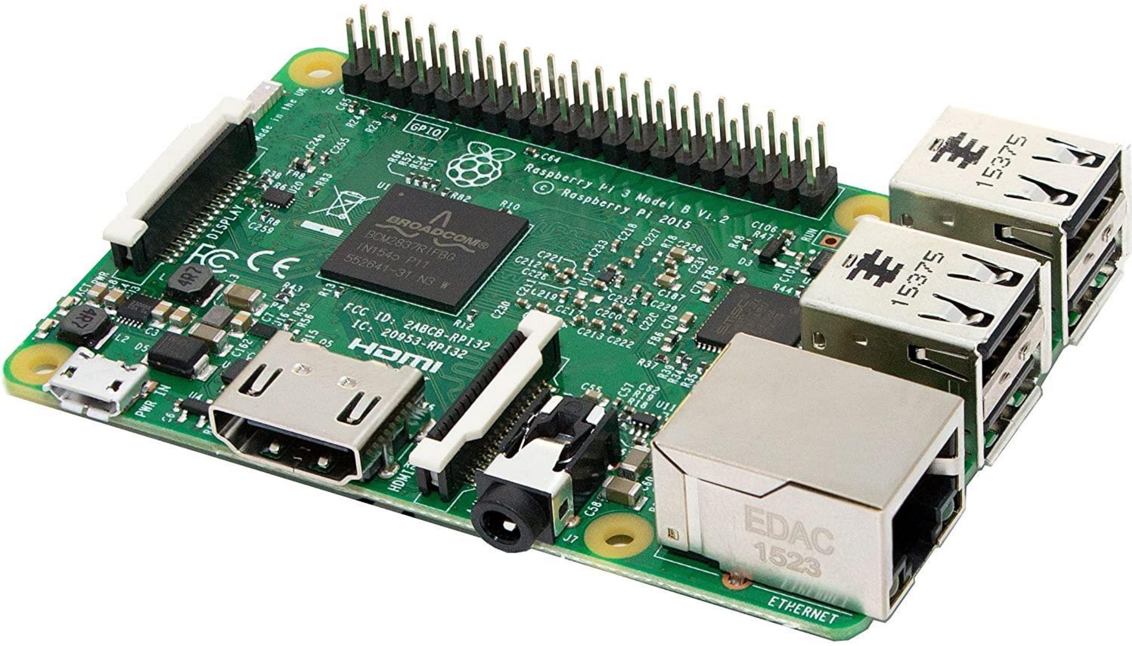


**Figure 26: Raspberry Pi 3 model b**

## B-1.2 COMPONENTS REQUIRED TO CONFIGURE THE RASPBERRY-PI

Before the raspberry pi can be implemented, it has to be configured by installing an operating system and other software utilities. An internet connection (mobile phone hotspot) is required along with the following components: a laptop, an SD card and a USB SD card adapter. The operating system of the raspberry pi is first downloaded from the raspberry pi website and then installed into the SD Card which is inserted into the laptop via the SD Card reader. The SD Card is then inserted into the raspberry-pi computer through the micro SD Card slot. After that, the installation of the operating system is done automatically when the pi is powered.

## B-1.3 THE CAMERA

The USB webcam will be used for this project. It will be connected to a USB port of the raspberry pi.

**Figure 27: The USB webcam connected to the raspberry pi through the USB port**

## B-1.4 ULTRASONIC SENSOR

The ultrasonic sensor works almost on the same principle as a radar system. The transmitter emits a high frequency ultrasonic sound, which bounces off any nearby solid objects. Some of that ultrasonic noise is reflected and detected by the receiver on the sensor. That returned signal is then processed by the control circuit to calculate the time difference between the signal being transmitted and received. This time can subsequently be used, along with some clever math, to calculate the distance between the sensor and the reflecting object. The ultrasonic sensor module-HC-SR04 is used for this project. This sensor calculates the distance between itself and the object detected by the deep learning algorithm.

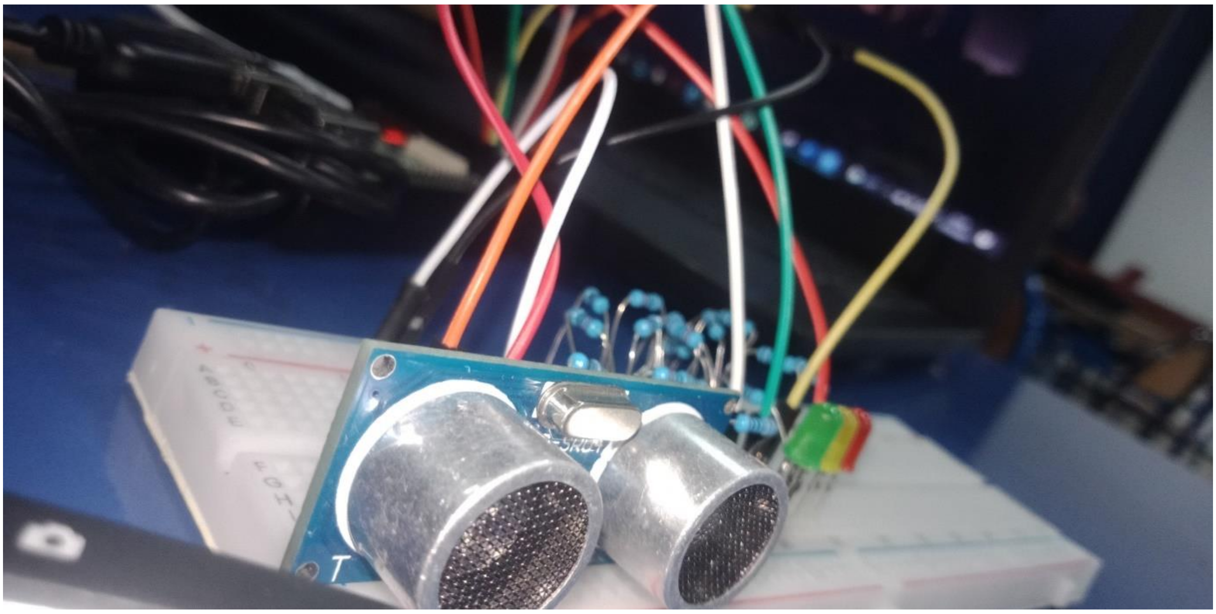


**Figure 28: HC-SR04 Ultrasonic Sensor distance sensor**

## B-1.5 BREADBOARD, LEDS, RESISTORS AND JUMPER WIRES

A breadboard is a thin plastic board used to hold electronic components (transistors, resistors, chips etc.) that are wired together. It's used to develop prototypes of electronic circuits. The breadboard is used here along with resistors, jumper wires and LEDs.

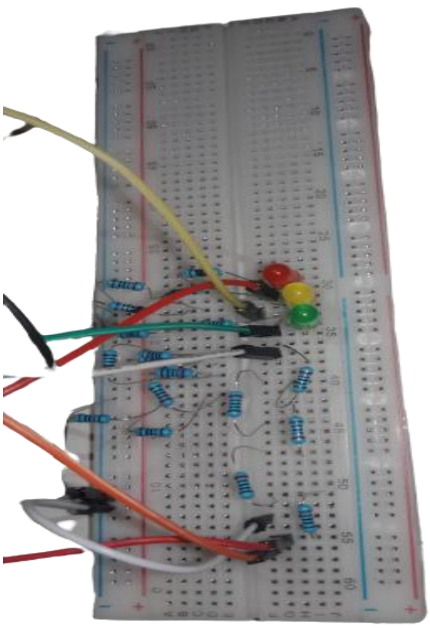


**Figure 29: LEDs, resistors and jumper wires in the breadboard**

## B-2 SUMMARY OF THE PROPOSED ARCHITECTURE

After seeing all the devices that we need to build the prototype of the smart obstacle detection system, it's essential to know how they are going to interact together. Hence, after configuring the raspberry pi and linking it with the camera, the deep learning algorithm is then implemented into it. The ultrasonic sensor is also inserted in the breadboard along with some resistances to adapt and connect it to the pi. The architecture is then composed and ready to be powered.

When powered, the prototype will be able to do the following tasks:

- Detect obstacles in real time along with their corresponding bounding boxes.
- Classify the detected obstacles according to their various classes.
- calculate the distance of the detected obstacles with respect to the position of the ultrasonic sensor.
- Display all of the above information on the screen of the computer (used as monitor).
- Offer warning indications according to the threat level of collision using LEDs.

# SECTION C: PRESENTATION OF THE DATASET AND THE NEURAL NETWORK ALGORITHM DEVELOPED

# C-1 PRESENTATION OF THE DATASET

## C-1.1 DEFINITION OF DATASET AND WHY IT IS USEFUL FOR THE PROTOTYPE

A dataset is defined as a collection of data that is treated as a single unit by a computer. This means that a dataset contains a lot of separate pieces of data but can be used to train an algorithm with the goal of finding predictable patterns inside the whole dataset. Also, data is an essential component of any AI model and its availability in large amounts is one of the reasons why deep learning took off in the recent years. Usually, a single dataset is split into three parts: a training set, a validation set and a test set. In other cases, it gets split into just two: a training set and a test set. Defining these two terms, a training set is a dataset of examples used during the learning process while a test set is the dataset used to evaluate the accuracy of the trained algorithm.

## C-1.2 PRESENTATION OF THE DATASET FOR THE PROTOTYPE

Two types of datasets are presented and one of them is used for the prototype in section-d of part3 (COCO).

- <u>COCO Dataset</u>

  For a real-time object detection with some objects which are familiar to the environment, a dataset taken from the COCO database can be used. The Common Objects in Context (COCO) dataset is one of the most popular open source object recognition databases used to train deep learning programs. This database includes hundreds of thousands of images with millions of already labeled objects for training. However, the problem of this dataset is that it only permits us to classify all specific objects into a general group that they belong to. For example: a tanker, a cargo ship and a passenger ship will all be classified simply as “ship” or “boat. (Dataset available at:[43])

- <u>Ship-only Dataset from KAGGLE</u>

  In this case, a ship-only dataset is taken from KAGGLE for ship classification. KAGGLE is an online community of data scientists and machine learning practitioners. It allows users to find and publish datasets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers and enter competitions to solve data science challenges. This dataset allows us to classify ships to their specific categories. It contains 6252 images in the training set and 2680 in the test set. The categories of ships that can be classified and their corresponding codes in the dataset are as follows: Cargo ships as class 1, Military ships class 2, Aircraft Carrier class 3, Cruise ships class 4 and Tankers as class 5. (Dataset available at:[44])

# C-2 THE NEURAL NETWORK ALGORITHM DEVELOPED FOR THE PROTOTYPE

## C-2.1 NEURAL NETWORK ALGORITHM TRAINED WITH THE COCO DATASET

The algorithm developed for the prototype is downloaded from the TensorFlow1 Detection Model Zoo [45]. It provides a collection of detection models pre-trained on different datasets. In our case we take a model called SSDlite_MOBILENET_V2_COCO which is pre-trained with the coco dataset. To use this model, in addition to TensorFlow and OpenCV, another library has to be downloaded – Google Protocol Buffers (Protobuf). Protobuf is useful for developing programs to communicate with each other over a network or for storing data. The following are the categories of objects provided by the COCO dataset [43] that can be detected and classified by the algorithm:

```
{1: 'person',
 2: 'bicycle',
 3: 'car',
 4: 'motorcycle',
 5: 'airplane',
 6: 'bus',
 7: 'train',
 8: 'truck',
 9: 'boat',
 10: 'traffic light',
 11: 'fire hydrant',
 13: 'stop sign',
 14: 'parking meter',
 15: 'bench',
 16: 'bird',
 17: 'cat',
 18: 'dog',
 19: 'horse',
 20: 'sheep',
 21: 'cow',
 22: 'elephant',
 23: 'bear',
 24: 'zebra',
 25: 'giraffe',
 27: 'backpack',
 28: 'umbrella',
 31: 'handbag',
 32: 'tie',
 33: 'suitcase',
 34: 'frisbee',
 35: 'skis',
 36: 'snowboard',
 37: 'sports ball',
 38: 'kite',
 39: 'baseball bat',
 40: 'baseball glove',
 41: 'skateboard',
 42: 'surfboard',
 43: 'tennis racket',
```

```
44: 'bottle',
46: 'wine glass',
47: 'cup',
48: 'fork',
49: 'knife',
50: 'spoon',
51: 'bowl',
52: 'banana',
53: 'apple',
54: 'sandwich',
55: 'orange',
56: 'broccoli',
57: 'carrot',
58: 'hot dog',
59: 'pizza',
60: 'donut',
61: 'cake',
62: 'chair',
63: 'couch',
64: 'potted plant',
65: 'bed',
67: 'dining table',
70: 'toilet',
72: 'tv',
73: 'laptop',
74: 'mouse',
75: 'remote',
76: 'keyboard',
77: 'cell phone',
78: 'microwave',
79: 'oven',
80: 'toaster',
81: 'sink',
82: 'refrigerator',
84: 'book',
85: 'clock',
86: 'vase',
87: 'scissors',
88: 'teddy bear',
89: 'hair drier',
90: 'toothbrush'}
```

## C-2.2 NEURAL NETWORK DEVELOPED FOR THE SHIP-ONLY DATASET

The ship-only dataset described above is not designed for object detection rather for object classification. However, we will integrate it in this section since an algorithm for ship classification is one step toward the development of a real ship detection system. In this regard, we partially trained the dataset in Google Colaboratory using a pre-trained CNN classifier called Xception model. Below is an overview outlining the training procedure.

- Upload the dataset to Google Colaboratory

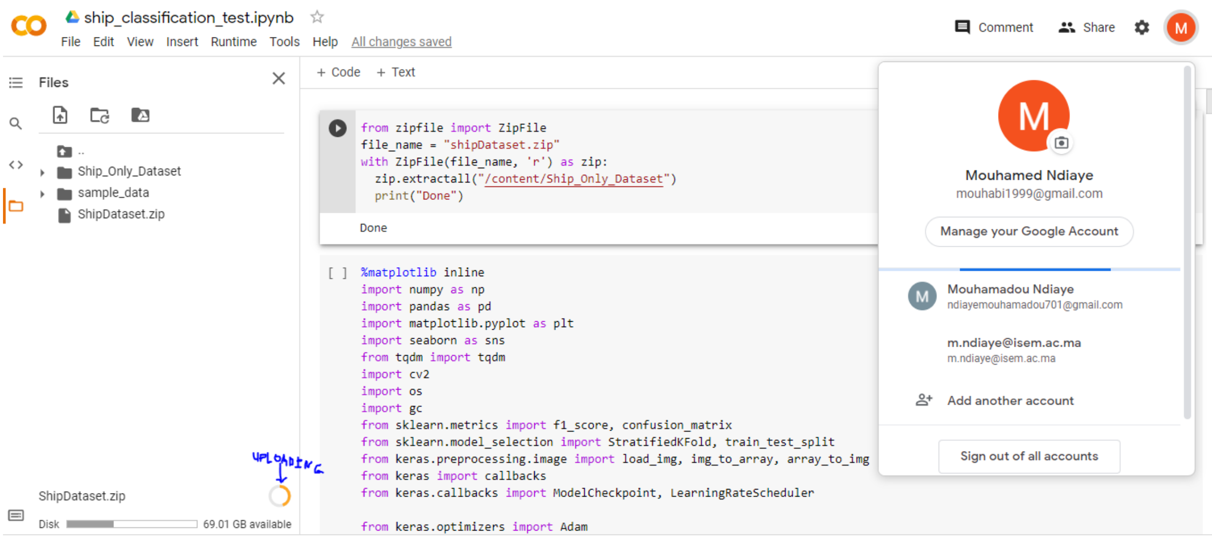


**Figure 30: Uploading the dataset to the Google Collaboratory notebook [authors]**

➢ Unzip the uploaded file and import all the dependencies

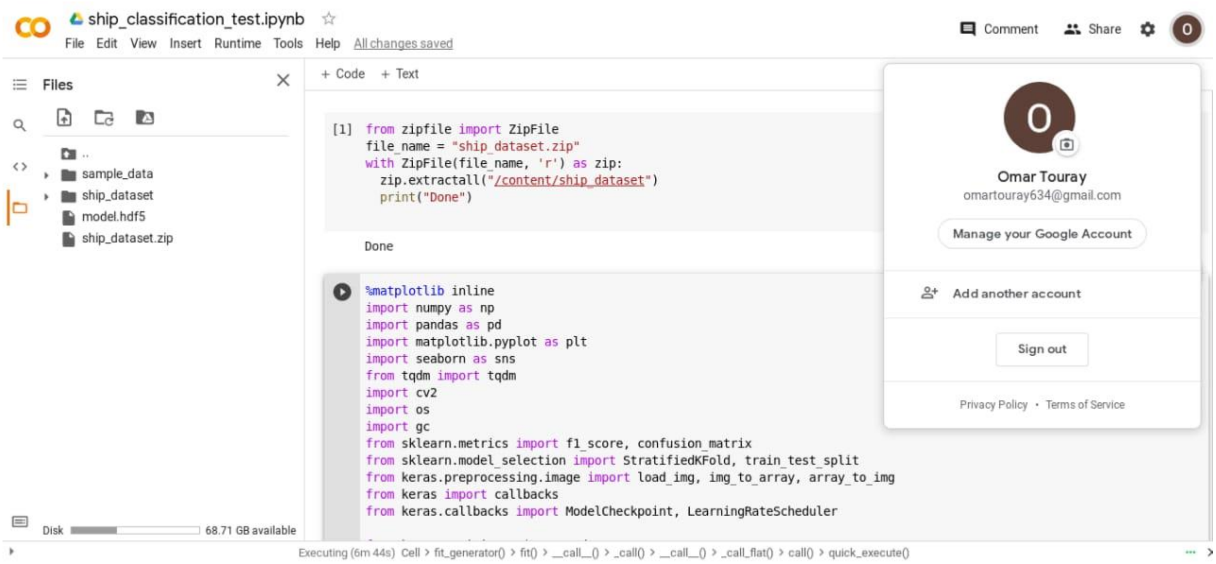


**Figure 31: unzipping the uploaded file and import all the dependencies [authors]**

➢ Load the test and the training set

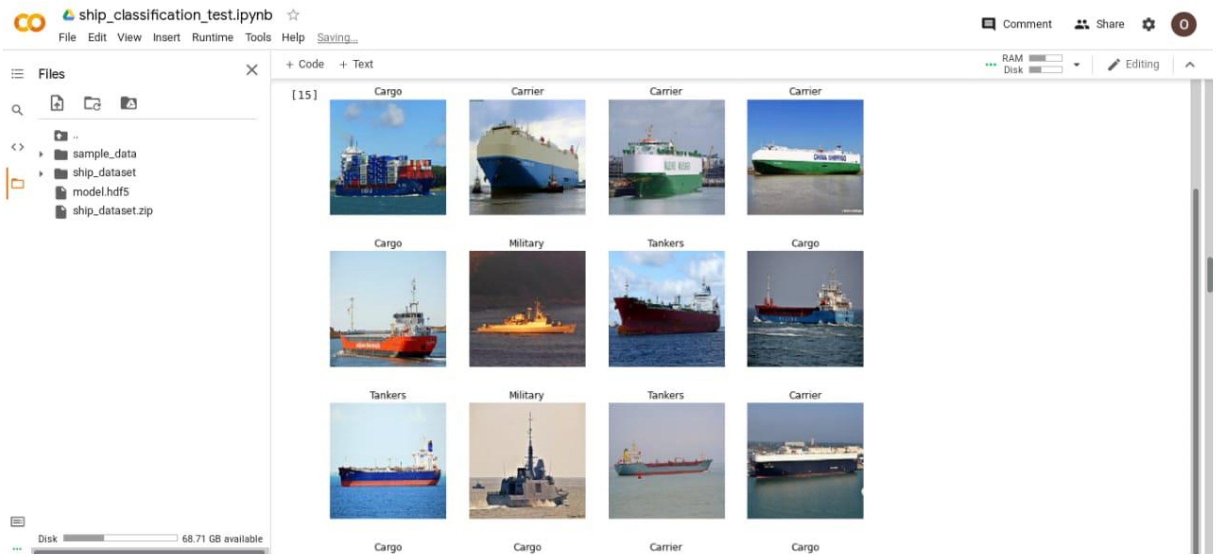


**Figure 32: Visualization of the train images [authors]**

- Import the pre-trained model (Xception)

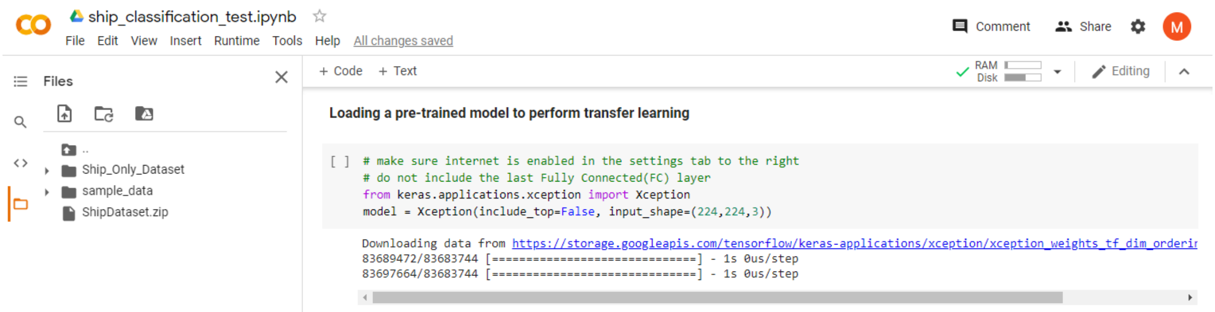


**Figure 33: importation of the pre-trained model used [authors]**

- Train the Xception model with the ship dataset

**Figure 34: Training of the model with the ship dataset [authors]**

# SECTION D: SIMULATION AND SOLUTION PROTOTYPING

This section is dedicated to the description of every concrete thing used in the realization of the Smart Obstacle Detection System. In this way, the following sub-sections describe every step taken for its prototyping.

## D-1 CONFIGURATION OF THE RASPBERRY PI

The raspberry pi provides computing power and interfacing for all the other components and therefore, needs to be configured to meet the needs of the system.

### D-1.1 INSTALLATION OF THE OPERATING SYSTEM (OS) IN THE SD-CARD

The operating system is installed in the SD-card inserted in the computer by intermediate of a USB SD-card adapter before its implementation in the pi. The following steps are followed for this operation.

- Format the SD-card using the application SD Memory Card Formatter (downloading available at:[46])

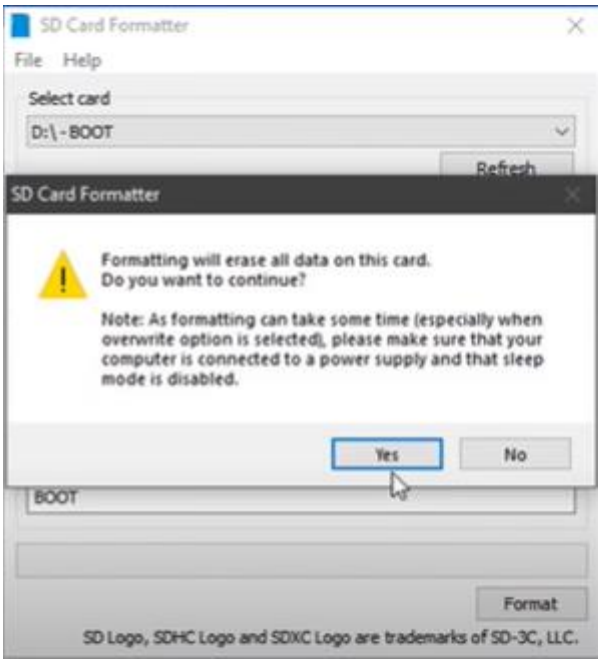



**Figure 35 : SD Memory Card Formatter interface**

- Download the raspbian operating system from the raspberry pi website

  *(Available At: [47])*

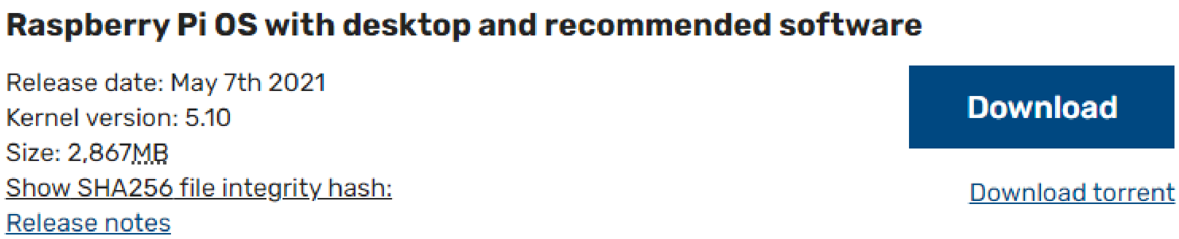



**Figure 36: Image of the Raspbian operating system to download**

- Upload the OS in the SD-card with the help of a software application called ETCHER *(Available at:[48])*

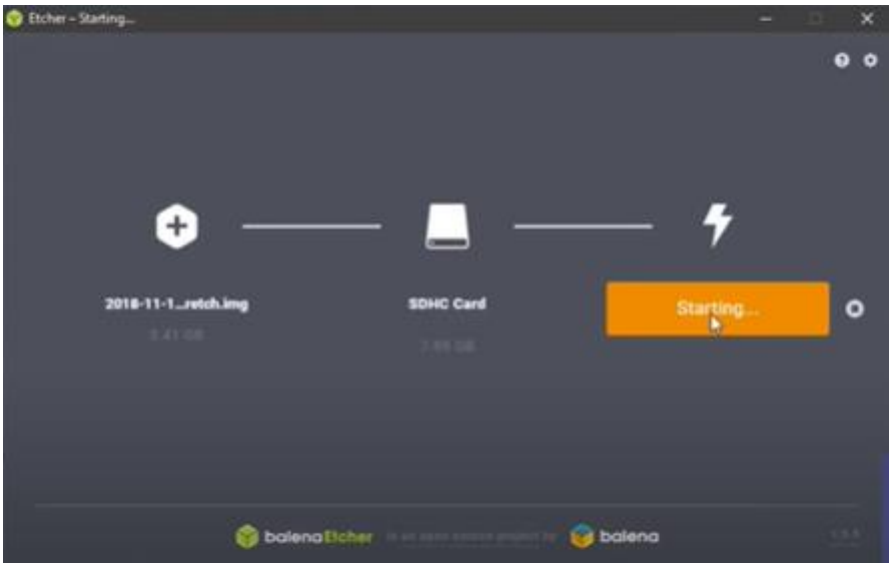
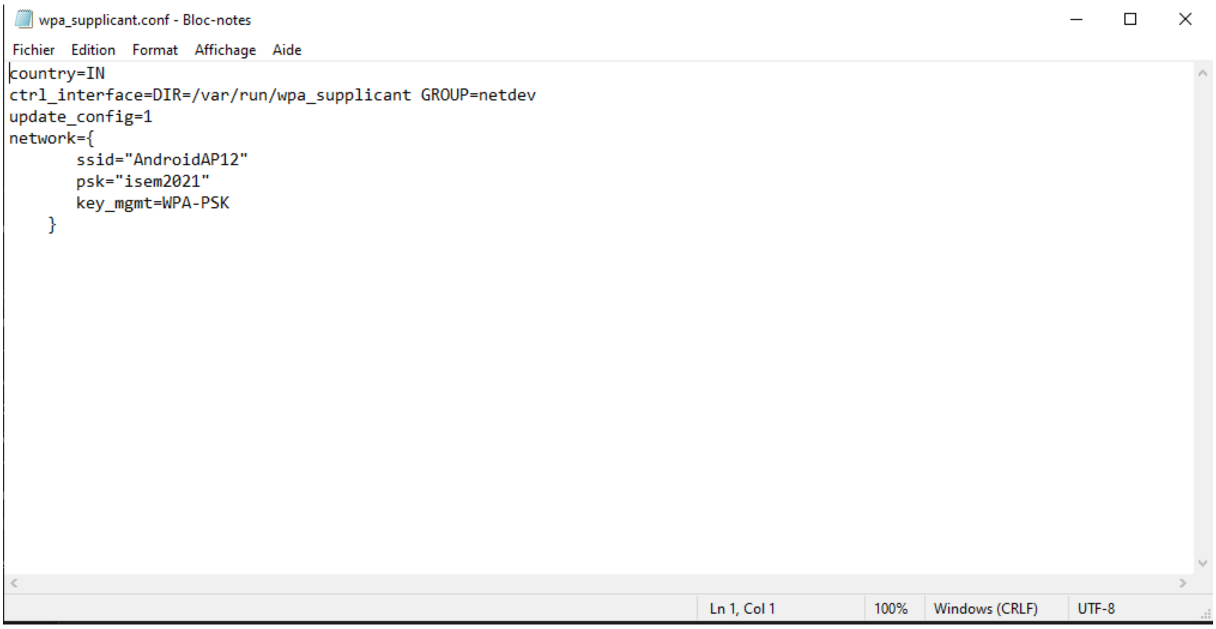



**Figure 37: Interface of the ETCHER application**

## D-1.2 CONNECTION OF THE RASPBERRY PI TO THE COMPUTER

This connection is done wirelessly. To achieve this, the Pi and the computer are connected in the same Local Area Network (LAN) using a mobile hotspot. A file in the raspberry pi SD-card is configured in such a way that it contains the name and the password of the mobile phone hotspot.




```
country=IN
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
network={
        ssid="AndroidAP12"
        psk="isem2021"
        key_mgmt=WPA-PSK
    }
```

**Figure 38: Text file configured for the connection of the pi to the mobile Hotspot named AndroidAP12**

The following software applications are also required for the remote connection of the raspberry pi.

➢ ADVANCED IP SCANNER

*(Available at: [49])*

This is a desktop application that allows us to capture the IP address of the raspberry pi when it's connected to the hotspot in the same way as the computer.

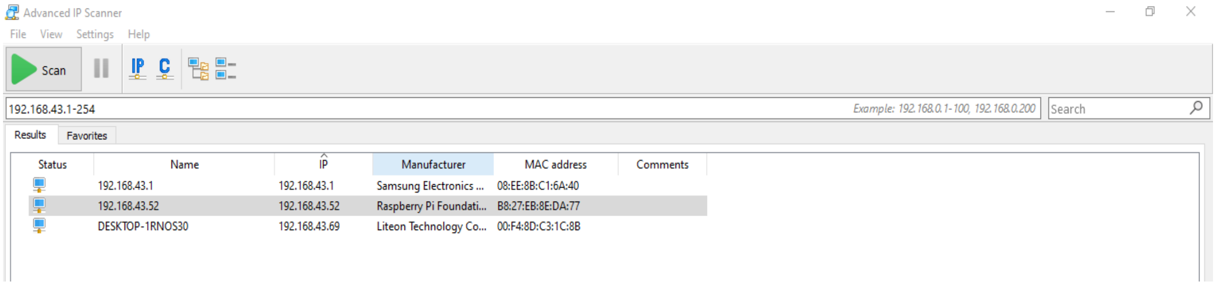



**Figure 39: Interface of the Advanced Ip Scanner software**

➢ PUTTY

(Available at: [50])

Putty is a desktop application which permits us to connect the raspberry pi remotely through SSH (Secure Shell). The IP address of the raspberry pi captured using the advanced IP scanner is pasted in the putty application to allow us to access to the terminal of the raspberry pi and configure the tools that we need in it.

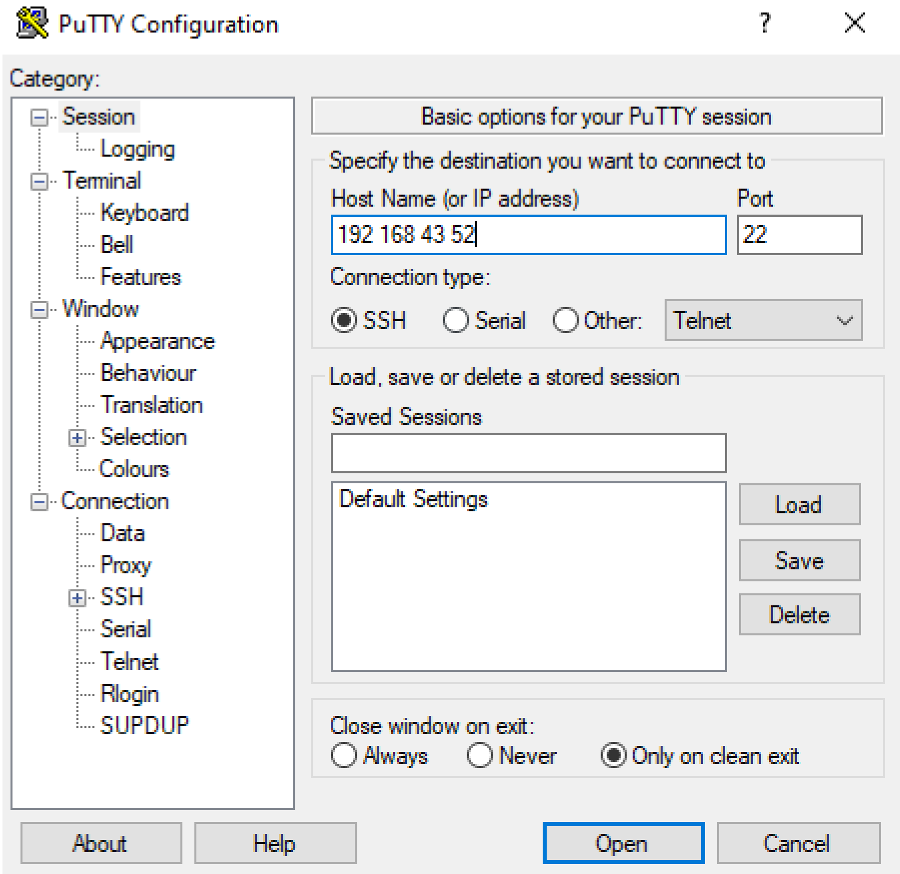



**Figure 40: Interface of the PUTTY software**

- VNC VIEWER
  *(Available at: [51])*

After enabling the VNC server of the raspberry pi through the PUTTY, the VNC application then allows us to connect directly to the raspberry pi using its IP address.

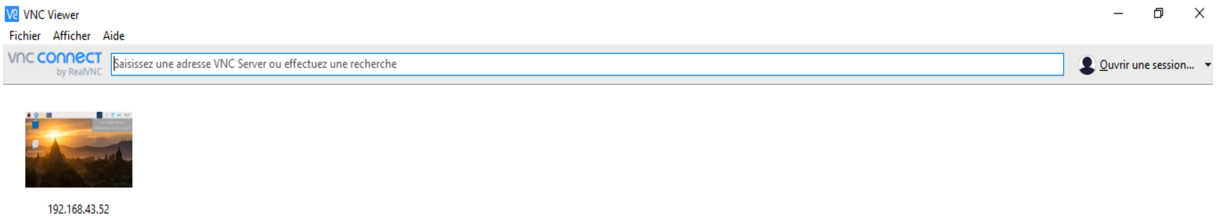



**Figure 41: Interface of the VNC Viewer**

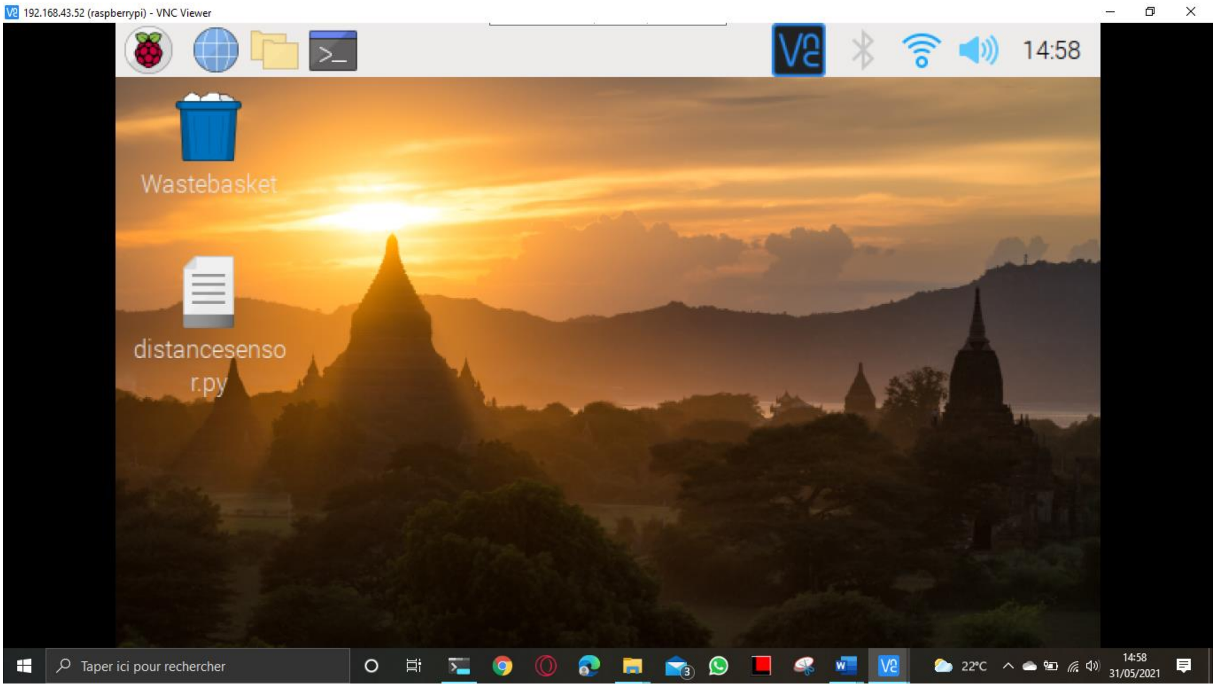



**Figure 42: Pi Desktop**

# D-2 IMPLEMENTATION OF THE NEURAL NETWORK ALGORITHM IN THE RASPBERRY PI

The implementation of the neural network algorithm is divided into seven steps. For each of the steps, we used the raspberry pi terminal as a development environment.

(NB: the pictures in black are the commands executed on the Pi terminal)

## D-2.1 UPDATE OF THE RASBERRY PI

The raspberry pi needs to be fully updated and upgraded at first.

```
pi@raspberrypi:~ $ sudo apt-get update
Hit:1 http://raspbian.raspberrypi.org/raspbian stretch InRelease
Hit:2 http://archive.raspberrypi.org/debian stretch InRelease
Reading package lists... Done
pi@raspberrypi:~ $
pi@raspberrypi:~ $
pi@raspberrypi:~ $ sudo apt-get dist-upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libpng-tools lxkeymap python-cairo python-gobject python-gobject-2
  python-gtk2 python-xklavier
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  chromium-browser-l10n chromium-codecs-ffmpeg-extra
The following packages will be upgraded:
  bluez-firmware chromium-browser gnupg gnupg-agent libmad0
  libpackagekit-glib2-18 libprocps6 libwavpack1 lxplug-network lxplug-volume
  nodered packagekit pi-bluetooth pi-package pi-package-data
  pi-package-session piclone procps python-blinkt python-phatbeat
  python-rainbowhat python3-blinkt python3-phatbeat python3-rainbowhat
  raspberrypi-ui-mods raspi-config rc-gui rpi-chromium-mods wget xdg-utils
```

# D-2.2 INSTALLATION OF TENSORFLOW

The downloading of the library is rather large (over 100MB), and the following command is executed for that:

```
pip3 install tensorflow
```

Some other packages that are essential for TensorFlow Object Detection are also installed through these commands:

```
sudo apt-get install libatlas-base-dev
```

```
sudo pip3 install pillow lxml jupyter matplotlib cython
sudo apt-get install python-tk
```

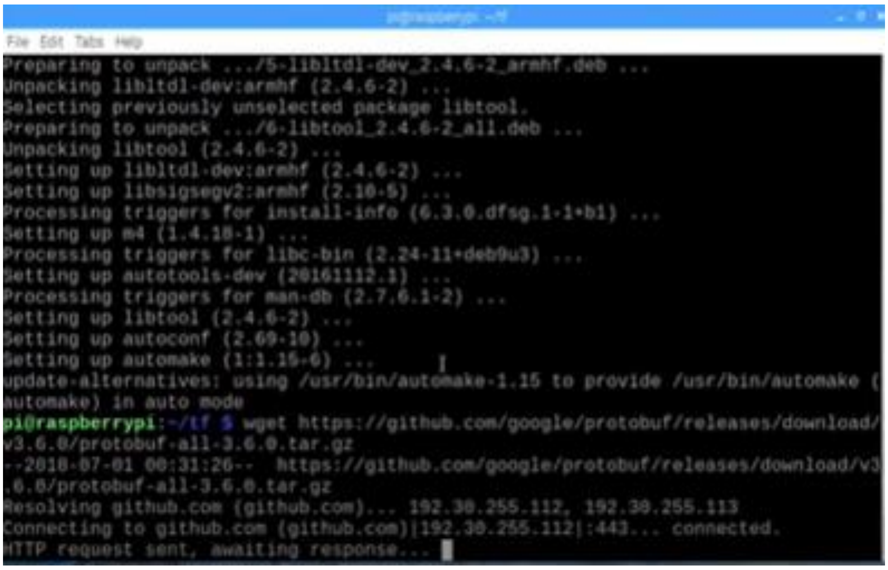
# D-2.3 INSTALLATION OF OPENCV

```
sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev
sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev
sudo apt-get install libxvidcore-dev libx264-dev
sudo apt-get install qt4-dev-tools libatlas-base-dev
```

```
sudo pip3 install opencv-python
```

## D-2.4 COMPILATION AND INSTALLATION OF PROTOBUF

```
sudo apt-get install protobuf-compiler
```




## D-2.5 SET UP OF THE TENSORFLOW DIRECTORY STRUCTURE

```
mkdir tensorflow1
cd tensorflow1
```

```
git clone --depth 1 https://github.com/tensorflow/models.git
```

```
sudo nano ~/.bashrc
```

```
export PYTHONPATH=$PYTHONPATH:/home/pi/tensorflow1/models/research:/home/pi/tensorflow1/models/research/slim
```

```
cd /home/pi/tensorflow1/models/research
protoc object_detection/protos/*.proto --python_out=.
```

```
cd /home/pi/tensorflow1/models/research/object_detection
```

## D-2.6 DOWNLOAD OF THE SSDlite_MOBILENET_V2

```
wget http://download.tensorflow.org/models/object_detection/ssdlite_mobilenet_v2_coco_2018_05_09.tar.gz
tar -xzvf ssdlite_mobilenet_v2_coco_2018_05_09.tar.gz
```

## D-2.7 EXECUTION OF THE PROGRAM IN THE PI TERMINAL

At first, a python file is downloaded (we named it Object_detection_picamera.py):

```
wget https://raw.githubusercontent.com/EdjeElectronics/TensorFlow-Object-Detection-on-the-Raspberry-Pi/master/
```

And then the script is run through the USB webcam with this command:

```
python3 Object_detection_picamera.py --usbcam
```

RESULTS:

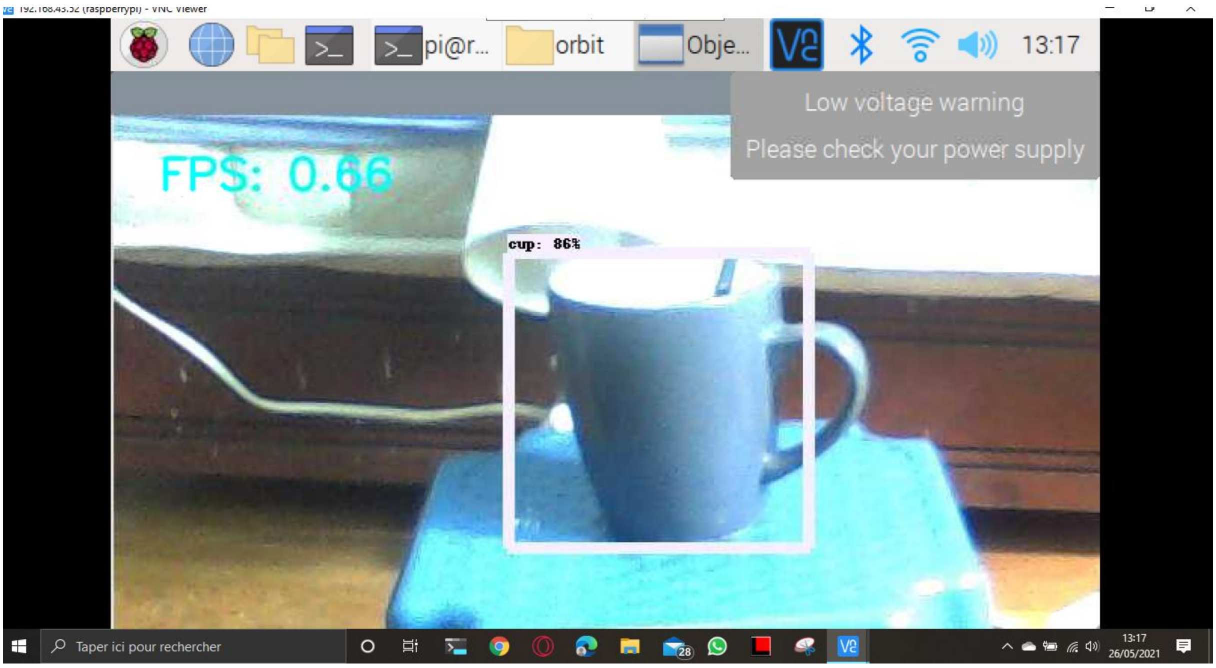



**Figure 43: Detection of a cup with an accuracy of 86%**

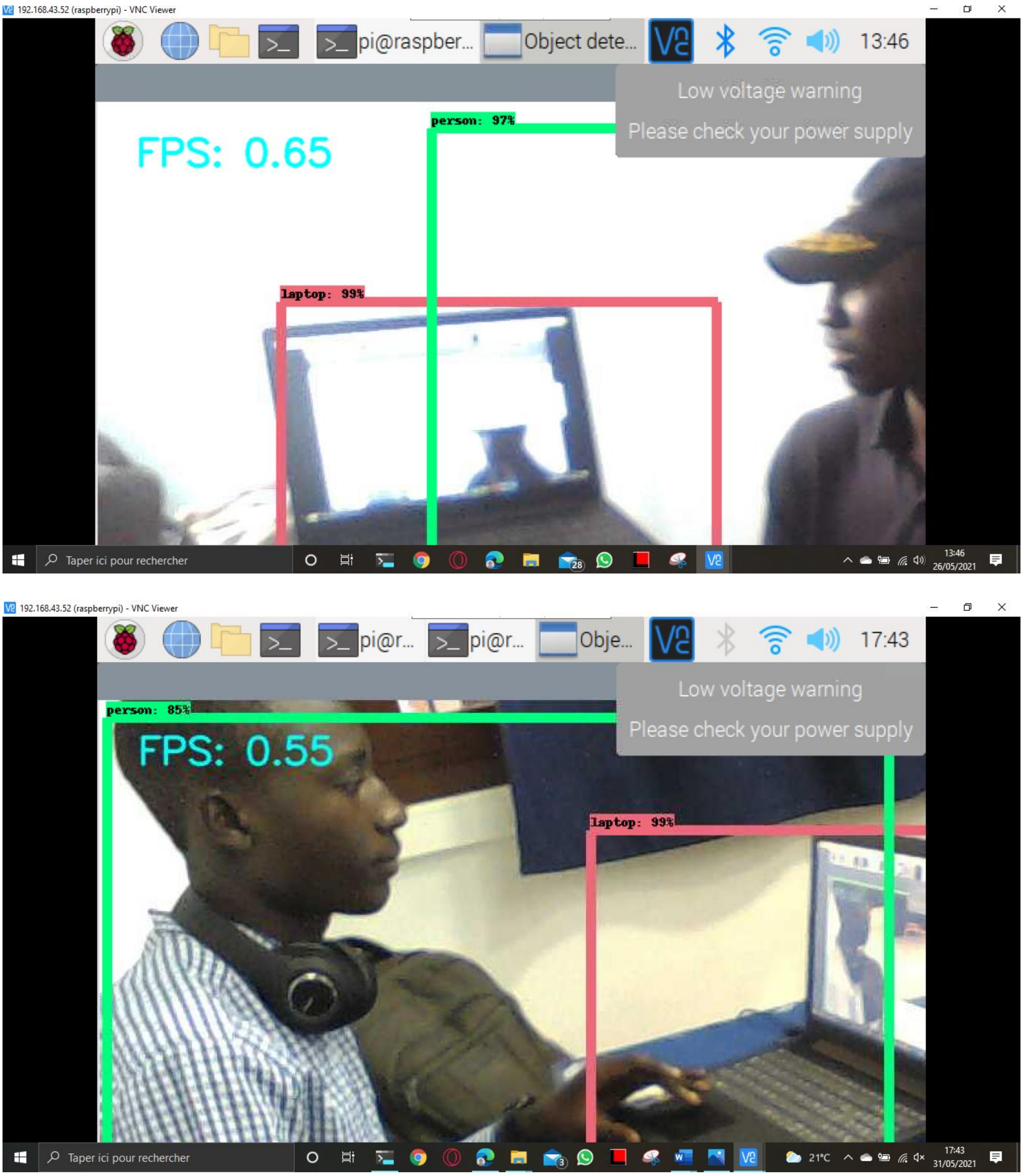



**Figure 44: Detection of the Authors and their respective laptops by the Algorithm**

# D-3 CONNECTION OF THE COMPONENTS OF THE SYSTEM

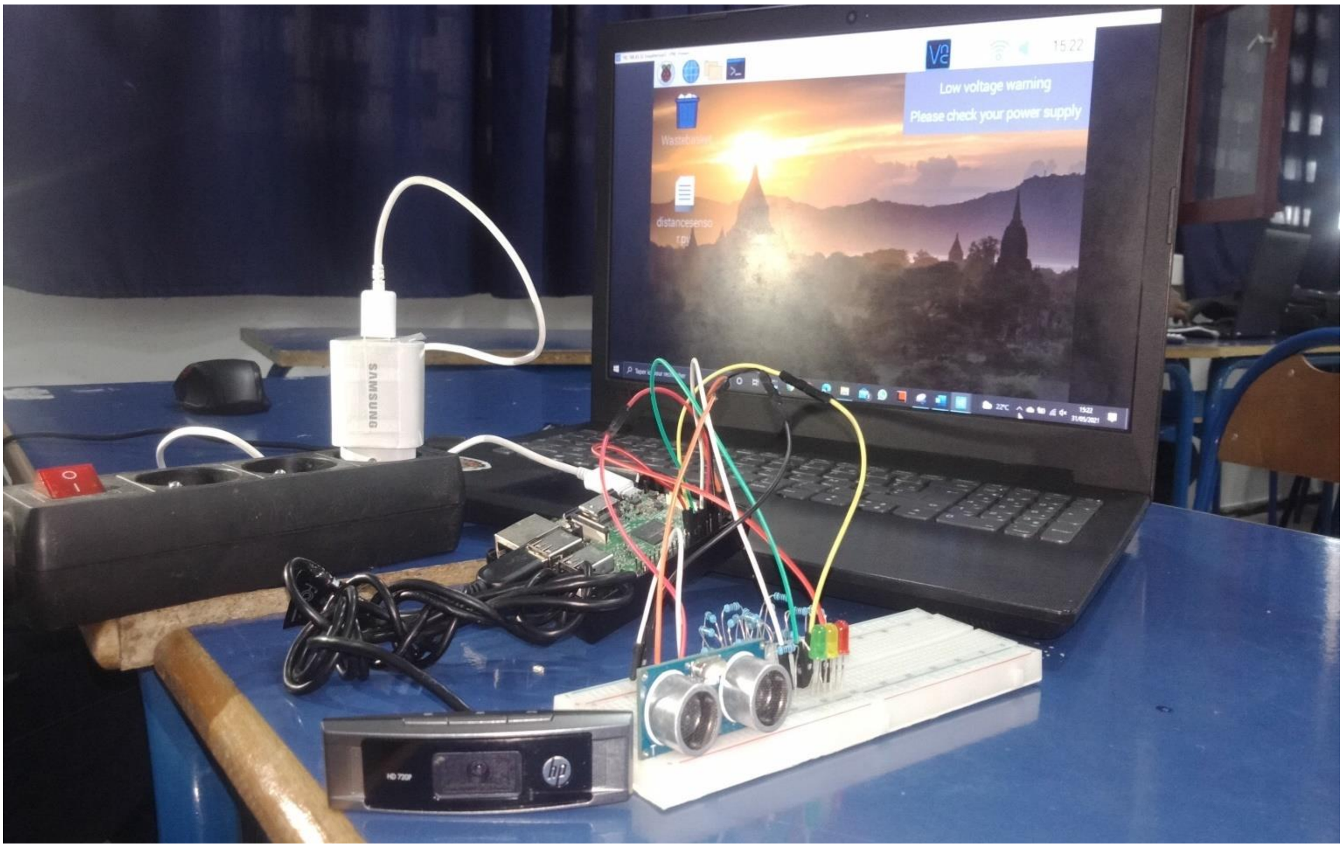



**Figure 45: Connection of the components of the system**

# D-4 CONFIGURATION AND EXECUTION OF THE ULTRASONIC SENSOR PROGRAM ALONG WITH THE LEDs

```
import RPi.GPIO as GPIO
import time

GPIO.setwarnings(False)
GPIO.cleanup()

GPIO.setmode(GPIO.BCM)
TRIG = 4
ECHO = 18

GREEN = 17
YELLOW = 27
RED = 22

GPIO.setup(TRIG,GPIO.OUT)
```

```
GPIO.setup(ECHO,GPIO.IN)

GPIO.setup(GREEN,GPIO.OUT)
GPIO.setup(YELLOW,GPIO.OUT)
GPIO.setup(RED,GPIO.OUT)

def green_light():
    GPIO.output(GREEN, GPIO.HIGH)
    GPIO.output(YELLOW, GPIO.LOW)
    GPIO.output(RED, GPIO.LOW)

def yellow_light():
    GPIO.output(GREEN, GPIO.LOW)
    GPIO.output(YELLOW, GPIO.HIGH)
    GPIO.output(RED, GPIO.LOW)

def red_light():
    GPIO.output(GREEN, GPIO.LOW)
    GPIO.output(YELLOW, GPIO.LOW)
    GPIO.output(RED, GPIO.HIGH)

def get_distance():

    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)

    while GPIO.input(ECHO) == False:
        start = time.time()

    while GPIO.input(ECHO) == True:
        end = time.time()

    sig_time = end-start
    distance = sig_time / 0.000058
    return distance

while True:
    distance = get_distance()
    time.sleep(0.05)
    print(distance)

    if distance >= 30:
        green_light()
    elif 30 > distance > 10:
        yellow_light()
    elif distance <= 10:
        red_light()
```

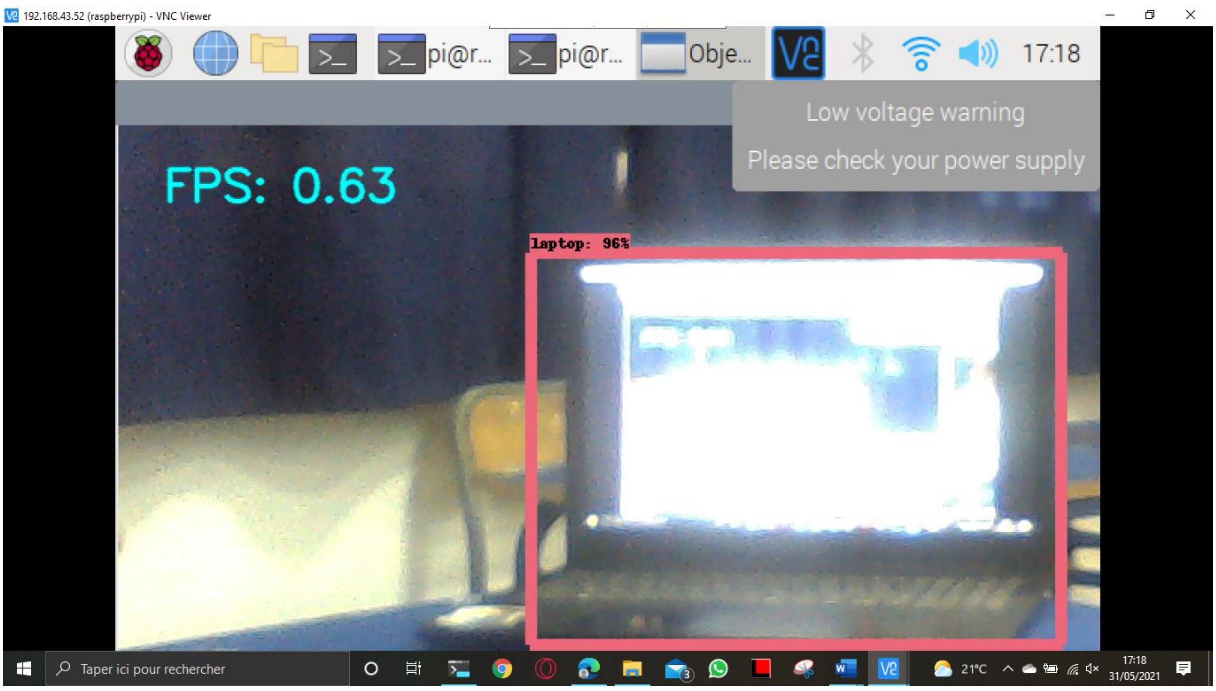
# D-5 EMULATION OF THE SMART OBSTACLE DETECTION SYSTEM

After the configuration and connection of all the components as well as the execution of their corresponding programs, the system is now tested all together and the following tasks are realized concurrently and in real-time:

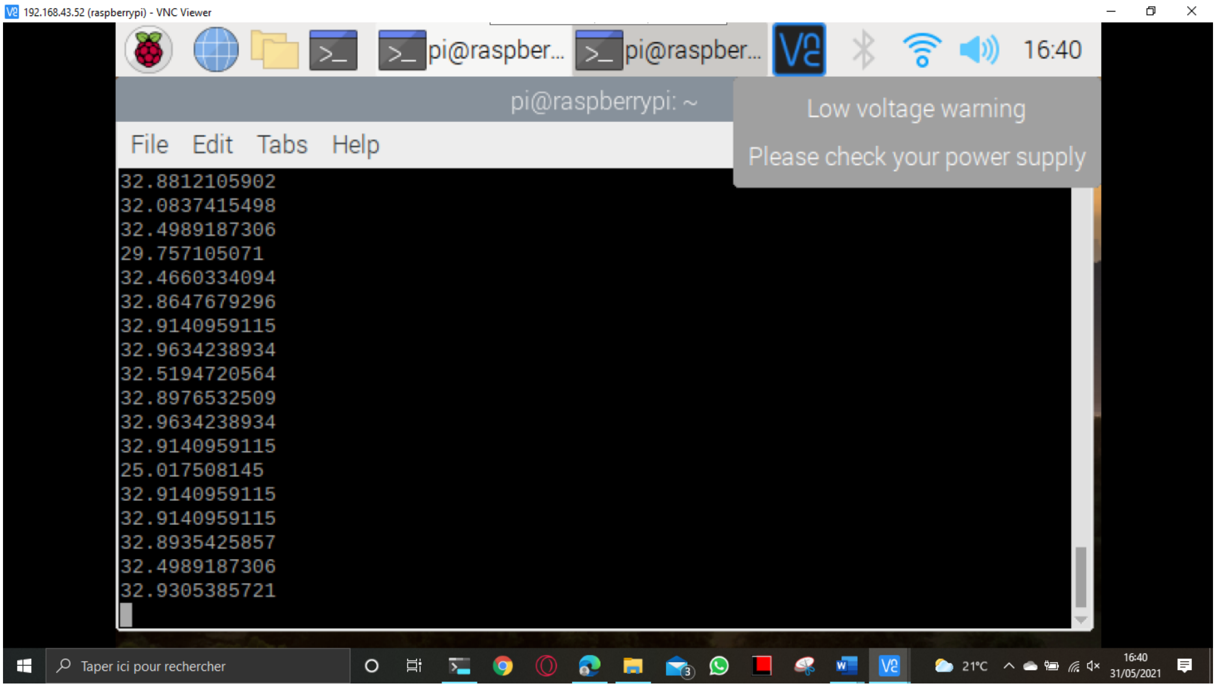
- Detection of obstacles within the camera range along with their corresponding bounding boxes.
- Classification of the detected obstacles according to their various classes.
- Determination of the distance of the detected obstacle from the ultrasonic sensor.
- Allocation of a warning signal based on the proximity of the obstacle using different LED lights.
- Display of all the above information in the screen of the computer used as a monitor for the Pi

NB: During the emulation of the prototype, we choose a laptop as the obstacle that the deep learning algorithm has to detect

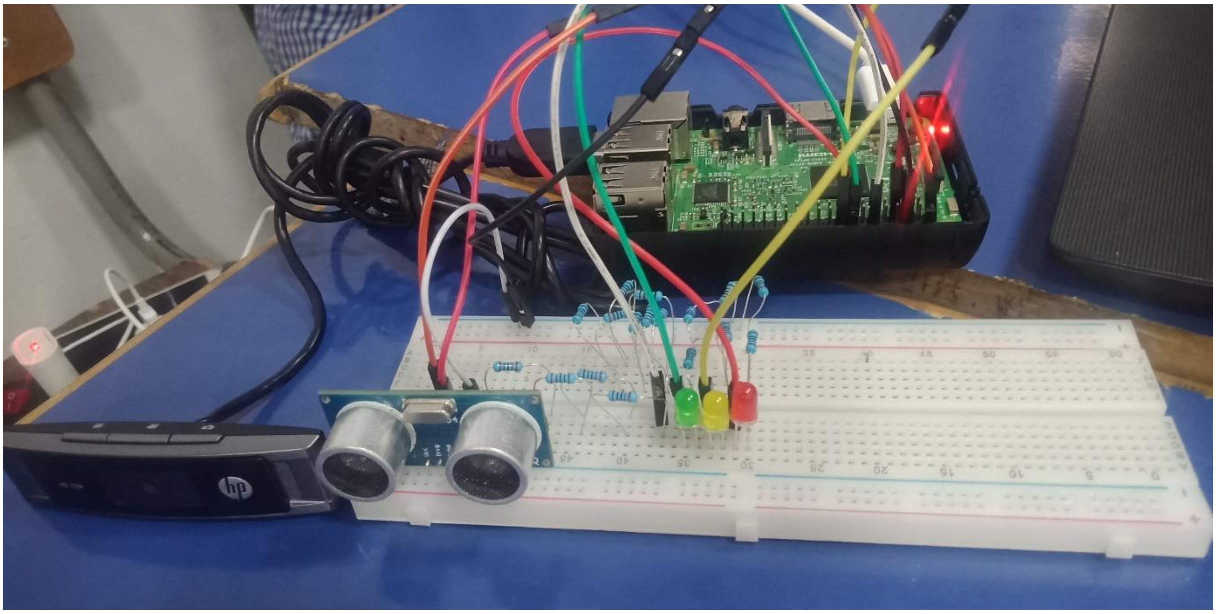
## a) DETECTION OF A LAPTOP SITUATED AT A DISTANCE MORE THAN 30 cm (WARNING: GREEN LED ON !!!)




Detection of the laptop done thanks to the deep learning algorithm with an accuracy of 96%

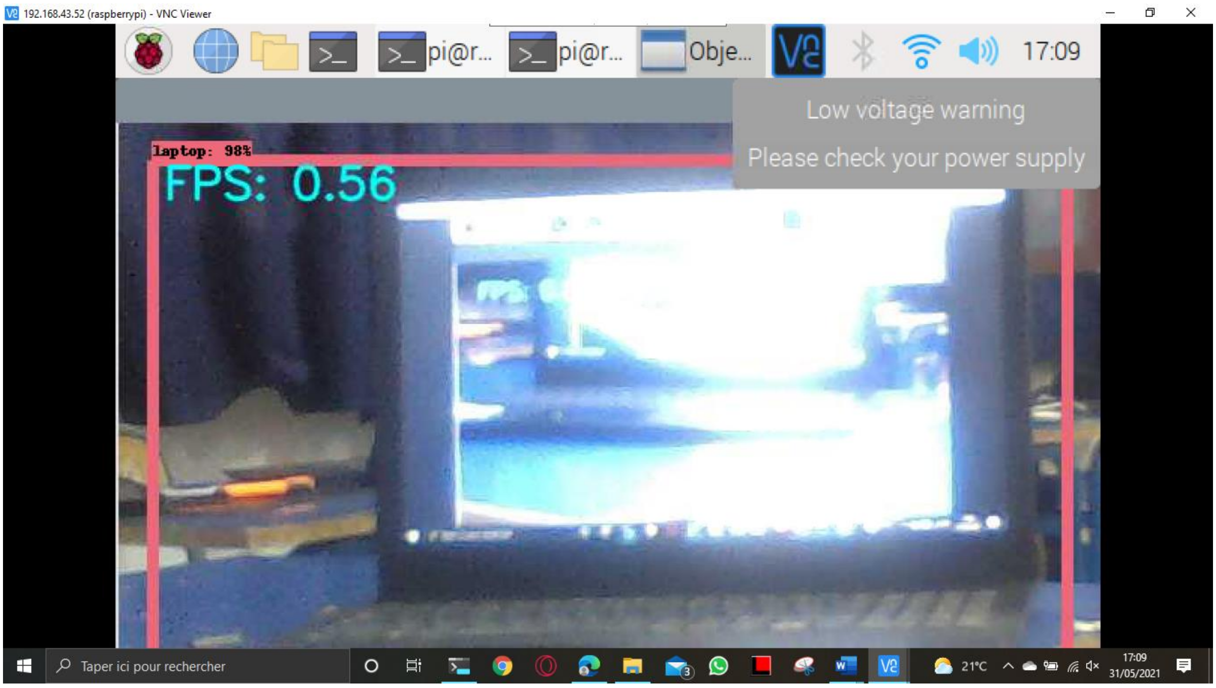



Measure of the distance of the laptop detected according to its position from the prototype thanks to the ultrasonic sensor program (distance measured: 32.93 cm)

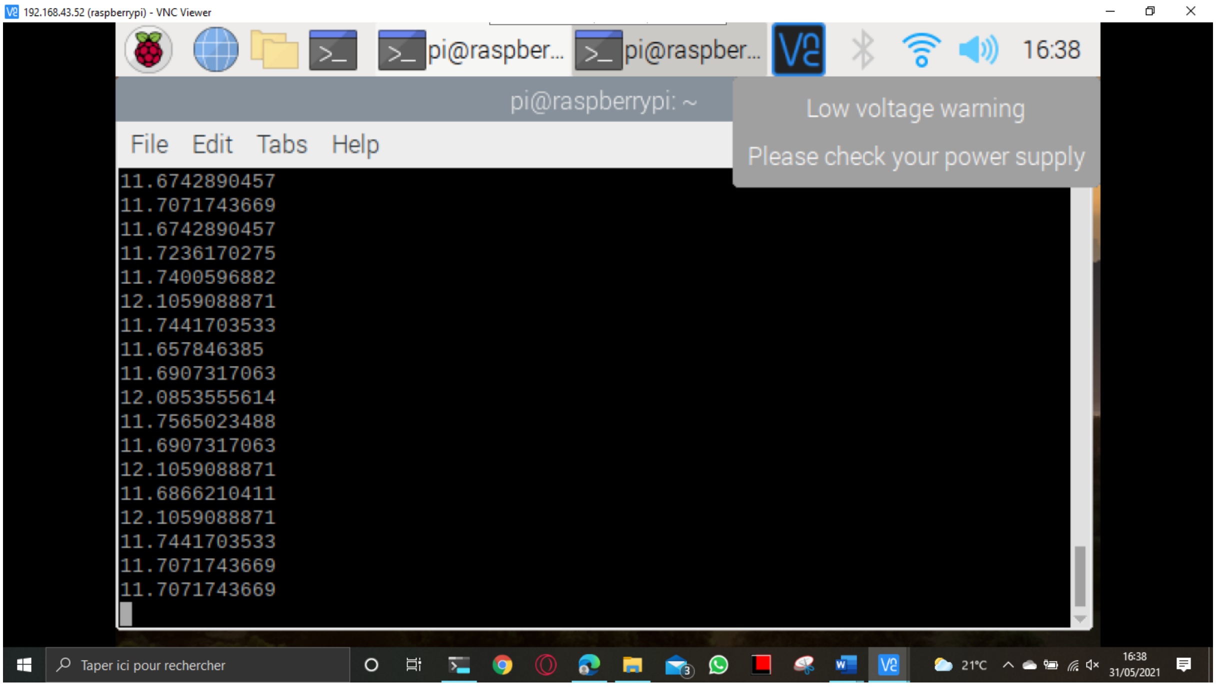


Warning system showing the green led on due to the position of the obstacle detected (>30cm)

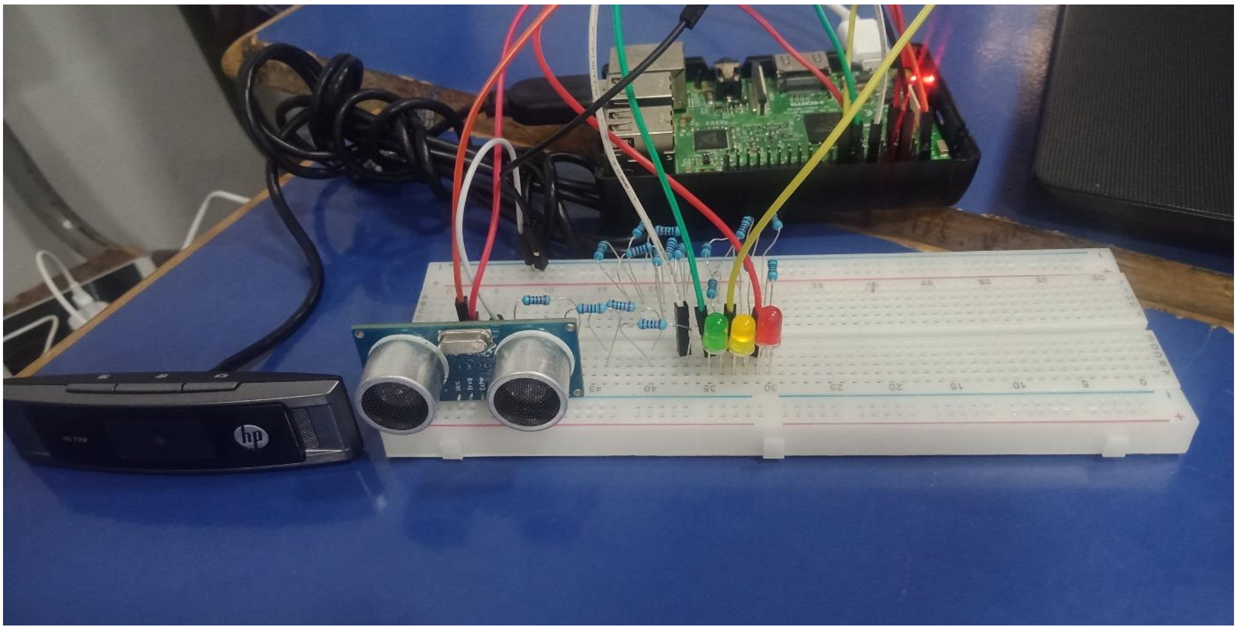
b) DETECTION OF A LAPTOP SITUATED AT A DISTANCE BETWEEN 10 AND 30 cm (WARNING: YELLOW LED ON !!!)




Detection of the laptop done thanks to the deep learning algorithm with an accuracy of 98%

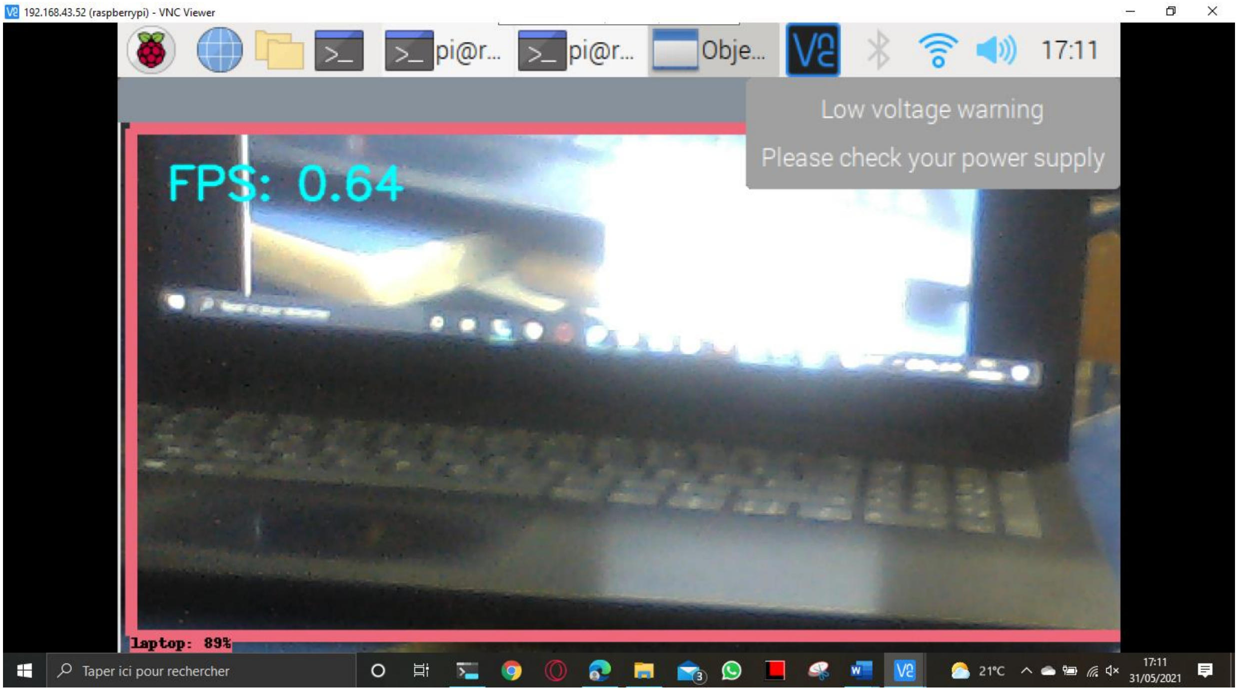



Measure of the distance of the laptop detected according to its position from the prototype thanks to the ultrasonic sensor program (distance measured: 11.7cm)

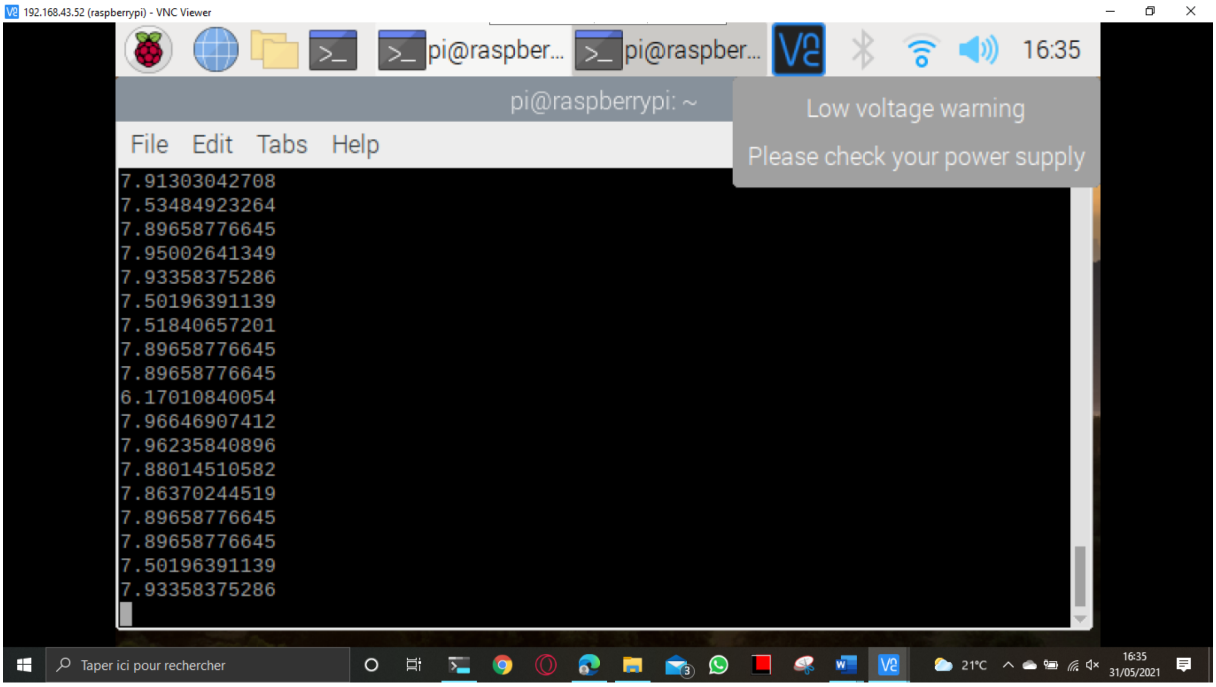


Warning system showing the yellow led on due to the position of the obstacle detected (10cm<p<30cm)

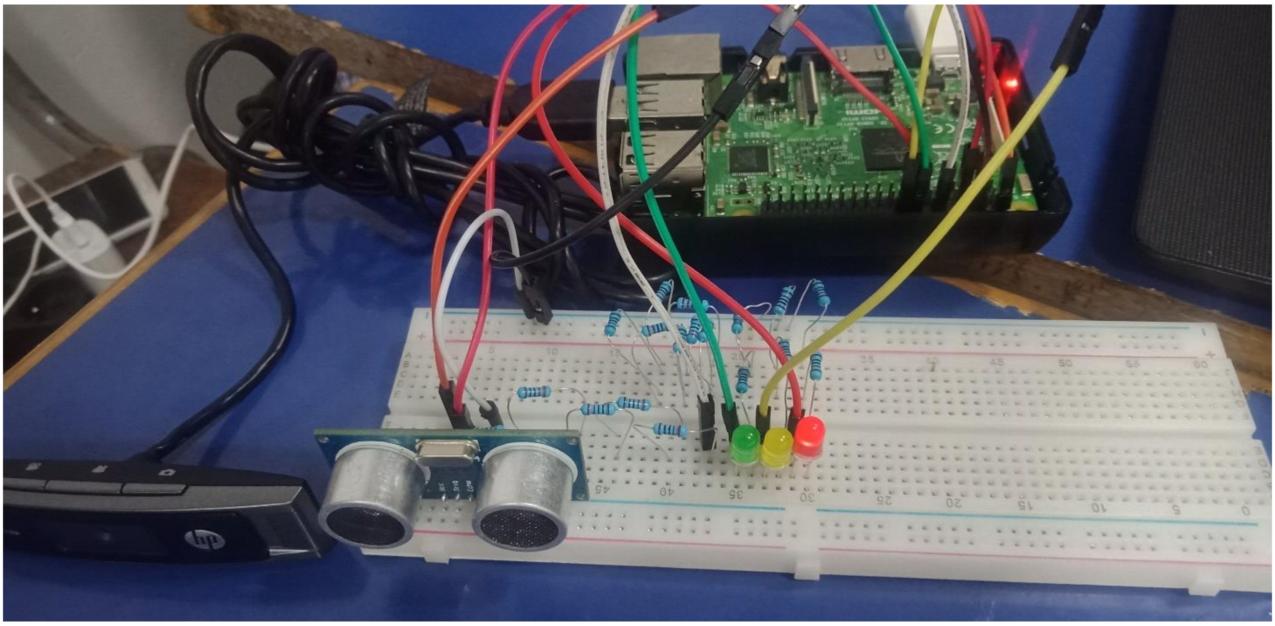
c) DETECTION OF A LAPTOP SITUATED AT A DISTANCE LESS THAN 10 cm (WARNING: RED LED ON !!!)




Detection of the laptop done thanks to the deep learning algorithm with an accuracy of 89%



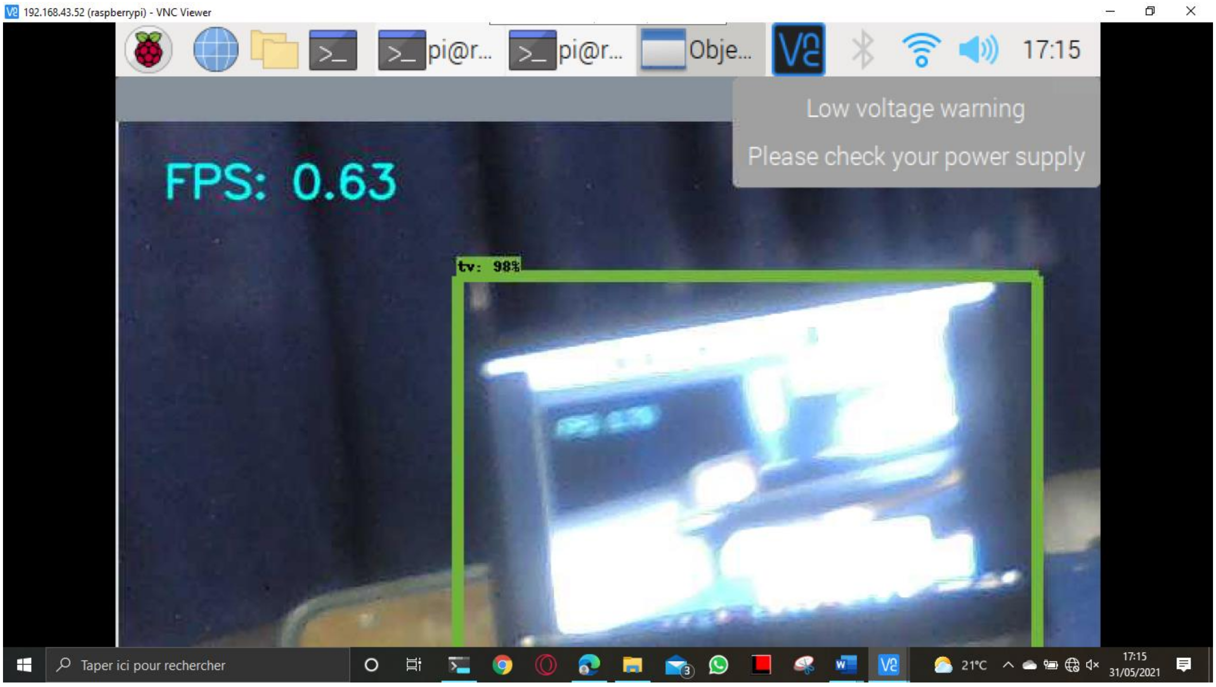

Measure of the distance of the laptop detected according to its position from the prototype thanks to the ultrasonic sensor program (distance measured: 7.93 cm)



Warning system showing the red led on due to the position of the obstacle detected (<10cm) (!!!danger)

However, the algorithm sometimes fails to detect some objects or make a mistake in the classification of the object.

As an example:




Here we can see that the algorithm confused the laptop for a TV with an accuracy of 98%

# CONCLUSION

One of the biggest lessons to learn from the previous industrial revolutions is that technology has the capability of revolutionizing and propelling humanity to an unimaginable level, not just through creating new fields of achievement but through the facilitation of human tasks and making them less risky and timely to accomplish.

An obstacle detection system plays a crucial role in maritime transport by extending the vision of the seafarers and helping them avoid collisions beforehand. However, with the traditional marine obstacle detection systems, the high risk of collision persists and gets more and more overwhelming as more and more ships are built to sail the seas. The solution to this increasing threat is to harness and integrate new technologies to the existing navigational tools. In regards to this, the tools of Artificial Intelligence and camera sensor fusion technology possess the necessary sophistication to help avoid or minimize collisions at sea especially during docking. As a result, the safety of the seafarers will be largely improved and better assured.

Indeed, the goal of this project was not to replace the technologies currently used on ships such as the radar/ARPA system, but to improve them to a point that the user (especially the seafarer) of these incumbent tools combined with our proposed system, acts more safely in the conduct of the vessel. Implementing an AI tool which is now one of the most revolutionary technologies, to try to improve safety in maritime traffic, was a great challenge that we wanted to raise up through this project. Thus, the project, in addition to allowing us to acquire new knowledge, opened us to a very large field that we have to keep informing ourselves, which is Artificial Intelligence.

A large part of this project has been dedicated toward the proposition of an independent obstacle detection system that is able to perform real time analysis of the environment and whose processed data will be used by the operator to make maneuvering decisions. Since one of the challenges of autonomous ships is having a sound obstacle detection system, don't we think that a research in the direction of this project is also a step toward tackling the autonomy of ships?

# REFERENCES:


[1] Radar and ARPA Manual: Radar and Target Tracking for Marine Radar Users, third edition 2013.
[2] https://blog.oureducation.in/duplexer/
[3] https://medium.com/@circleofmariners/imo-performance-standards-of-radar-5bd447c94af7
https://wwwcdn.imo.org/localresources/en/KnowledgeCentre/IndexofIMOResolutions/MSCResolutions/MSC.192(79).pdf
[4] https://www.sciencedirect.com/topics/engineering/marine-radar
[5] https://www.pfri.uniri.hr/bopri/documents/30_Marineradar_000.pdf
[6] https://lidarradar.com/info/errors-and-limitations-of-marine-radar
[7] https://lidarradar.com/info/advantages-and-disadvantages-of-radar-systems
[8] http://www.acceltex.com/planes-gains-and-beamwidths-reading-radiation-patterns/
[9]https://www.rfcafe.com/references/electrical/doppler.htm#:~:text=Doppler%20shift%20is%20an%20apparent,relative%20motion%20of%20two%20objects.&text=For%20a
[10] https://officerofthewatch.co.uk/2016/02/radar-initial-setup-performance-standards/
[11] https://en.wikipedia.org/wiki/S_band
[12] https://www.imo.org/en/KnowledgeCentre/IndexofIMOResolutions/Pages/msc-2004-06.aspx
[13] https://en.wikipedia.org/wiki/Anti-rolling_gyro
[14] https://www.verywellmind.com/theories-of-intelligence-2795035
[15] Introduction to Artificial Intelligence. Second Edition, 2017.
[16] https://www.britannica.com/technology/artificial-intelligence
[17] https://en.wikipedia.org/wiki/Connectionism
[18] https://iep.utm.edu/connect/
[19] https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/
[20] https://www.dataversity.net/brief-history-artificial-intelligence/
[21] https://www.networkworld.com/article/3534015/uk-startup-cisco-team-to-test-wi-fi-for-autonomous-vehicles.html
[22] https://hackernoon.com/blockchain-artificial-general-intelligence-benefit-humanity-230e213cae12
[23] https://www.forbes.com/sites/cognitiveworld/2019/06/19/7-types-of-artificial-intelligence/?sh=4a6a0fd233ee
[24] https://i.dailymail.co.uk/1s/2021/01/25/09/38442138-9182767-image-a-9_1611568096761.jpg
[25] https://www.analyticssteps.com/blogs/6-major-branches-artificial-intelligence-ai
[26] https://www.peculiar-coding-endeavours.com/2019/mlp_vs_cnn/
[27] https://towardsdatascience.com/power-of-a-single-neuron-perceptron-c418ba445095?gi=f7cfabe9c197
[28] https://laptrinhx.com/a-conversation-about-deep-learning-2962193597/
[29] https://www.peculiar-coding-endeavours.com/2019/mlp_vs_cnn/
[30] https://www.researchgate.net/figure/CNN-architecture_fig2_339905578
[31] https://www.sealevel.com/2020/10/06/types-of-neural-networks/
[32] https://medium.com/@tifa2up/image-classification-using-deep-neural-networks-a-beginner-friendly-approach-using-tensorflow-94b0a090ccd4
[33] http://datahacker.rs/deep-learning-object-localization/

[34] https://axonator.com/object-detection-technology
[35] http://deepmachinelearningai.com/object-tracking-in-deep-learning/
[36] https://www.intechopen.com/books/visual-object-tracking-with-deep-neural-networks/multi-person-tracking-based-on-faster-r-cnn-and-deep-appearance-features
[37] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi "You Only Look Once: Unified, Real-Time Object Detection", 2015. Available: https://arxiv.org/abs/1506.02640
[38] https://ieeexplore.ieee.org/document/8764393
[39] https://www.cvedia.com/algorithm/ship-detector/
[40] https://towardsdatascience.com/deep-learning-for-ship-detection-and-segmentation-71d223aca649
[41] R. Glenn Wright "Intelligent Autonomous Ship Navigation Using Multi-Sensor Modalities", 2019. Available: https://www.transnav.eu/Article_Intelligent_Autonomous_Ship_Navigation_Wright,51,923.html
[42] Felix Nobis, Maximilian Geisslinger, Markus Weber, Johannes Betz, Markus Lienkamp "A Deep Learning-based Radar and Camera Sensor Fusion Architecture for Object Detection", 2020. Available: https://arxiv.org/abs/2005.07431
[43]: https://cocodataset.org/#explore
[44]: https://www.kaggle.com/arpitjain007/game-of-deep-learning-ship-datasets
[45]:https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md
[46]: https://www.sdcard.org/downloads/formatter/
[47]: https://www.raspberrypi.org/software/operating-systems/#raspberry-pi-os-32-bit
[48]: https://www.balena.io/etcher/
[49]: https://www.advanced-ip-scanner.com/
[50]: https://www.putty.org/
[51]: https://www.realvnc.com/en/connect/download/viewer/

[Certificates earned from online courses by the authors in AI]