UNIVERSIDADE FEDERAL FLUMINENSE

ANDRÉ RIBEIRO BREITINGER

An Augmented Reality Periscope for Submarines with Extended Visual Classification

NITERÓI

2021

UNIVERSIDADE FEDERAL FLUMINENSE

ANDRÉ RIBEIRO BREITINGER

An Augmented Reality Periscope for Submarines with Extended Visual Classification

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação

Orientador:
Esteban Clua

NITERÓI

2021

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

B835a Breitinger, André Ribeiro
An Augmented Reality Periscope forSubmarines with Extended VisualClassification / André Ribeiro Breitinger ; Esteban Clua, orientador. Niterói, 2021.
63 f. : il.

Dissertação (mestrado)-Universidade Federal Fluminense, Niterói, 2021.

DOI: http://dx.doi.org/10.22409/PGC.2021.m.09947984745

1. Computer vision. 2. Mixed reality. 3. Submarine Periscope. 4. Synthetic data. 5. Produção intelectual. I. Clua, Esteban, orientador. II. Universidade Federal Fluminense. Instituto de Computação. III. Título.

CDD -

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

ANDRÉ RIBEIRO BREITINGER

An Augmented Reality Periscope for Submarines with Extended Visual Classification

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação

Aprovada em Outubro de 2021.

BANCA EXAMINADORA

Prof. Esteban Clua - Orientador, UFF

Prof. Leandro Augusto Frata Fernandes, UFF

Prof. Daniela Trevisan, UFF

Prof. Marcelo Knorich Zuffo, USP

Niterói

2021

ANDRÉ RIBEIRO BREITINGER

An Augmented Reality Periscope for Submarines with Extended Visual Classification

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação

Aprovada em Outubro de 2021.

BANCA EXAMINADORA

Prof. Esteban Clua - Orientador, UFF

Prof. Leandro Augusto Frata Fernandes, UFF

Prof. Daniela Trevisan, UFF

Prof. Marcelo Knorich Zuffo, USP

Niterói

2021

Dedico esse trabalho à toda minha família e amigos que sempre me apoiaram.

Acknowledgements

I want to thank my family for the support in times of absence, to the professors of the institute and to my advisor, Esteban Clua, for his dedication, support and patience. I also want to thank my colleagues and friends who supported me one way or another and helped me accomplish this challenge.

Resumo

Os submarinos são considerados extremamente estratégicos para qualquer força naval devido à sua capacidade furtiva. Emergir a superfície ou na cota periscópica é uma tarefa necessária para identificar os contatos visuais através do dispositivo do periscópio. Essa manobra possui muitos procedimentos e geralmente tem que ser rápida e ágil, para evitar a exposição. Nesta dissertação, apresentamos e implementamos uma nova arquitetura para periscópio de submarinos, desenvolvida para futuras operações da frota naval brasileira. Nosso sistema consiste em uma sonda que está conectada ao submarino e carrega uma câmara 360º. Projetamos e obtemos as imagens dentro da embarcação usando dispositivos VR / XR tradicionais. Também propomos e implementamos uma eficiente técnica de reconhecimento de objetos baseada em visão computacional usando imagens sintéticas, com o objetivo de estimar e exibir os navios detectados de forma eficaz e precisa. Para tanto, construímos e disponibilizamos um conjunto de dados composto por 99.000 imagens. Por fim, também estimamos as distâncias dos elementos classificados, mostrando todas as informações em uma interface baseada em AR. Embora a sonda seja conectada com fio, ela permite que a embarcação fique em posições profundas, reduzindo sua exposição e introduzindo uma nova forma de manobras submarinas.Validamos nossa proposta através de um experimento de experiência do usuário com 19 submarinistas especialistas em operações de periscópio.

Palavras-chave: computer vision, deep learning, mixed reality, object detection, periscope, synthetic data, submarine, transfer learning.

Abstract

Submarines are considered extremely strategic for any naval army due to their stealth capability. Periscopes are considered crucial sensors for the vessel and submerging emerging to the surface or periscope depth is a required task in order to identify visual contacts through this device. This maneuver has many procedures and usually has to be fast and agile, to avoid exposure. In this paper we present and implement a novel architecture for real submarine's periscopes, developed for future Brazilian naval fleet operations. Our system consists of a probe that is connected to the craft and carries a 360 camera. We project and take the images inside the vessel using traditional VR/XR devices. We also propose and implement an efficient Computer Vision-based MR technique to estimate and display detected vessels in an effective and precise way using synthetic images. For so, we built and make available a dataset composed of 99,000 images. Finally, we also estimate distances of the classified elements, showing all the information in an AR-based interface. Although the probe is wired-connected, it allows that the vessel stands in deep positions, reducing its exposure and introducing a new way for submarine maneuvers and operations. We validate our proposal through a user experience experiment using 19 experts in periscope operations.

Keywords: computer vision, deep learning, mixed reality, object detection, periscope, synthetic data, submarine, transfer learning.

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introduction

Brazil has an extensive maritime area, with unquestionable importance in different fields:

- One of main means of transport for the country's foreign trade;
- Diversity of natural resources such as fishing and marine biodiversity;
- Main oil and gas reserves and other mineral resources;
- Big Influence on the Brazilian and world climate.

As it has an area equivalent to 67% of our terrestrial territory, with a size and biodiversity similar to that of the Green Amazon, it was called the BLUE AMAZON [7].

The Brazilian sea holds immense reserves of oil and gas, in addition to other non-living resources (salt, gravel, sand, phosphorus, cobalt crusts, sulfides and poly metallic nodules, among others) that represent important sources of wealth for the country, in addition to contain a wide variety of marine organisms of biotechnological value that have properties with wide applications, mainly in the areas of pharmaceuticals, cosmetics, food and agriculture.

The Interministerial Commission for the Resources of the Sea - CIRM [8], guides the development of activities aimed at the effective use, exploration and sustainable use of natural resources in AMAZONIA AZUL and international areas, in accordance with the interests of Brazil and through its programs, encourages the training of human resources in the area of Marine Sciences, stimulates the development of research and innovation in different areas of knowledge, in addition to contributing to the expansion of a maritime mentality in the Brazilian population, arousing interest in the importance of the sea and the rational and sustainable use of its resources.

In addition to oil, the National Department of Mineral Production (DNPM) has already notified the Brazilian government of the potential for extracting metals with high economic valuen like nickel, copper, cobalt and manganese, located at great depths, around 4,000 meters.

The Navy maintains that, despite Brazil being in an area theoretically free of major conflicts, acting on the international scene based on the legitimacy given by International Government Organizations, requires efficient monitoring and surveillance. History shows that if a State has a valuable asset, over which there is imminent greed or demand from other actors, there is a situation of insecurity for this nation, which must surround itself with dissuasive means of power.

In this scenario, we propose a novel submarine monitoring and surveillance periscope, merging concepts of Augmented Reality, Computer Vision and 360 videos. Our proposal opens new possibilities for submarine activities.

1.1 Motivation

Submarines are among the most capable and strategic naval units to operate in areas where the enemy exercises some degree of control. The procedure adopted by many countries suggests that submarine actions are the priority in enemy monitoring, not only for reducing the control exercised by them but also for supporting other forces' actions. Also, the availability and presence of submarines significantly increase dissuasion potential due to the uncertainty of its actual position [11].

Submarines have their own operating characteristics and owe their special contribution to naval actions and operations to three intrinsic characteristics, known as basic features.

1. **Ability to Hide**: Which provides greater discretion in position and identification than to any other vessel, allowing the submarine to carry out its tactical actions in waters under enemy control.

2. **Relative Independence of Surface Environmental Problems**: which allows when the submarine operates in immersion, in adverse weather conditions, mainly regarding the state of the sea.

3. **Three-Dimensional Mobility**: which allows the submarine to explore the environmental conditions of sound propagation to carry out the attack, the evasion

and deception maneuvers, necessary to break the sonar contact, or even as an anti-torpedic measure, as can be seen in Fig. 1.1.

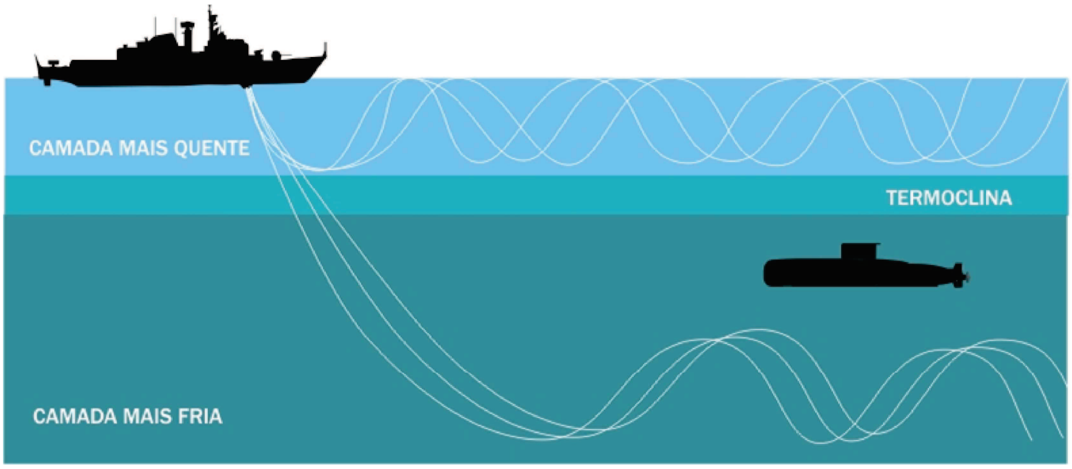


Figura 1.1: Sound propagation 3d exploration. Adapted from: [32]

One critical maneuver for submarines is the *periscope observation*, which requires the ship to navigate at periscope depth (Fig. 1.2). This exposition is strategically dangerous because the submarine can be detected by nearby enemies visually or by radar, becoming vulnerable. The periscope observation is made using a long periscope, a piece of optical equipment capable of rotating 360°, giving a panoramic view of the surface. Due to the degree of danger, this exposure should occur for just a few seconds and must be conducted by a trained officer operating the periscope which is assigned to identify contacts in visual range considered potential hazards during that short period.

Submarine Discretion Fee (SDF) is defined as a percentage ratio between the sum of the indiscretion periods (mast exposed) and the total submarine operation time. The objective of the submarine commander is to accomplish his mission as obtaining the minimum possible SDF.

The Brazilian navy adopts the technique called "perisher," which was developed by the British royal navy [6]. This technique was developed to maximize the amount of information obtained from the periscope while minimizing the exposure. Intermittent exposure reduce radar and visual detection probability. In the "perisher," the technique to perform a horizon scan takes 30 seconds, only to check if there is any hazard at the field of view, without any further observation of the detected contacts. A posterior investigation of each contact is made during 20 seconds on each one for identifying the elements, estimating the bow angle, measuring the distance with a stadiometer, and calculating the interval of observation, i.e., based on the contact's distance and its maximum speed, the

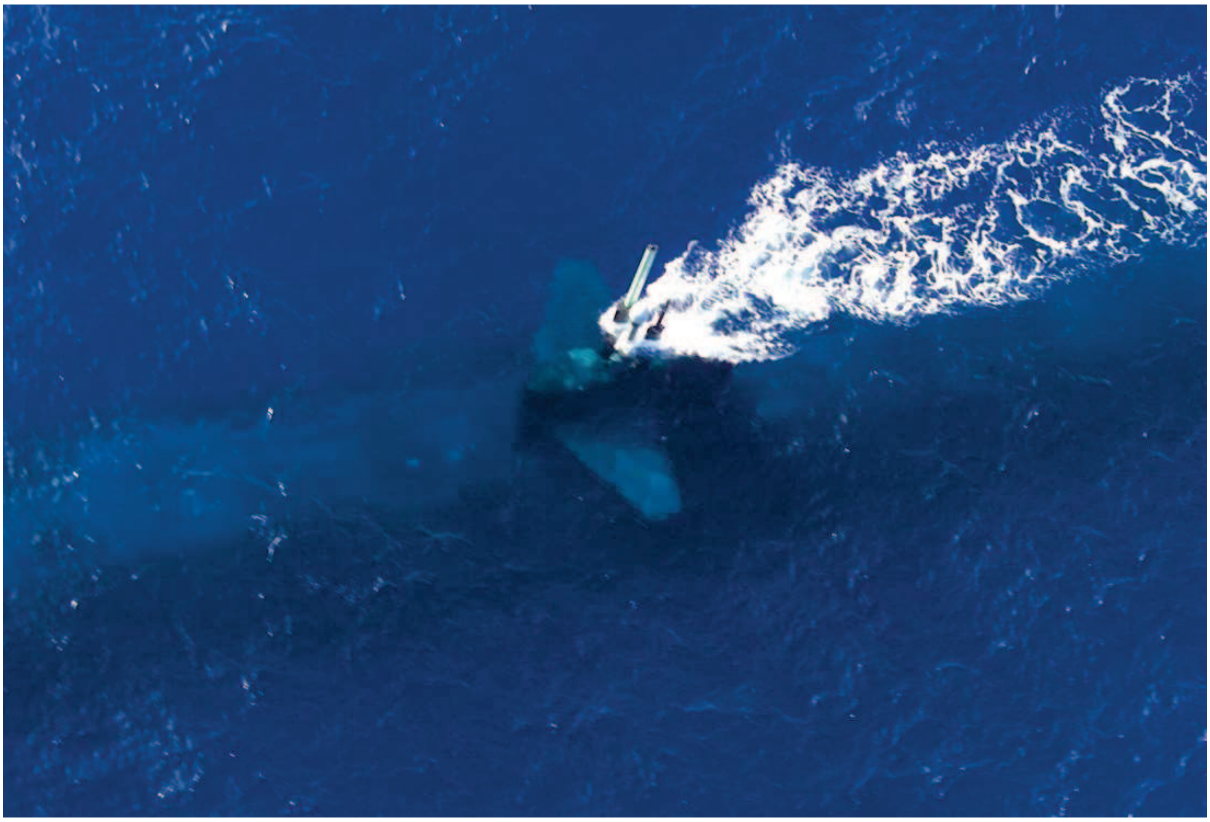


Figura 1.2: Periscope exposure at periscope depth [9].

periscope officer mentally calculates the maximum amount of time to observe the contact again putting the submarine in risk.

1.2 Objectives

The main objective of this dissertation is to propose a Augmented Reality (AR) periscope device, which is a novel and powerful solution capable of decreasing the periscope's exposure time and drastically increasing the observation tasks through Computer Vision techniques. Our solution is based on a wired probe that carries a high-resolution 360° camera and is connected to a commercial Head-Mounted Display (HMD) device, operated inside the submarine. We use different Computer Vision and Deep Learning techniques for surface elements classification and distance inference, which have potential to dismiss the use of conventional stadiometer requirement. This information is showed inside the HMD through Augmented Reality (AR) based interfaces, allowing fast and accurate decision making processes.

Deep Neural Networks (DNNs) have shown significant improvements in several application domains, including Image and Signal Processing. In Computer Vision, a specific type of DNN, known as Convolutional Neural Networks (CNNs), has revolutionized the state of the art of object detection and recognition, achieving faster and more accurate results [19].

We also propose the inclusion of different navigation information at the HMD display using Augmented Reality (AR) strategies.

We believe that our proposal will introduce a new way of operating periscopes and performing submarines operations in near future.

1.3 Contributions

Our main contributions can be summarized as:

- A proposal for a new architecture for submarines periscope using Augmented Virtuality HMD devices and approaches;
- A solution for ship classification in images taken from a periscope point of view;
- A ship distance estimation solution for recognized ships;
- An open dataset composed of 99,000 synthetic images of five (strategic) classes of ships.
- A user experience experiment that validates the usage of VR/AR devices for periscope operations.

1.4 Dissertation Organization

The chapters of this dissertation are divided as described below.

Chapter 2 we presents the correlating fields of knowledge that inspired our work, and discuss a similar solution also based on synthetic images for the training process.

Chapter 3 summarizes our proposal, presenting our novel periscope architecture, our proposed classification strategy, our object distance estimation approach, and our developed software functionality.

Chapter 4 shows our results and details about our training stage. It holds all experimental evaluation, associated studies, analysis and a user experience test with submarine officers. The complete questionnaire is at Appendix A.

Finally, in Chapter 5 we conclude our work and discusses future possibilities of our proposal.

Capítulo 2

Background and Related Works

As our work is multidisciplinary some concepts must be explained so that experts from different areas can be introduced to basics concepts from other areas to fully understand the dissertation, in the following sections we describe some of those concepts and correlate them with current works on the academy.

2.1 Background

2.1.1 Submarine and Navigation Issues

The International Regulations for Preventing Collisions at Sea (COLREGs) defines several rules to prevent collisions [13]. This is particularly dangerous to the submarine because there is a significant probability that other ships are not aware of the submarine's position.

Therefore, an extreme reality (XR) solution with a visual camera that can recognize dangerous elements from a deeper depth can cause a significant impact on navigation security.

Safety Quota is the minimum depth that will be safe for the submarine to travel when close to a surface unit, in order to ensure a separation between the ship's keel and the top of the submarine's sail, as can be seen at fig. 2.1

The techniques used for operation of the periscope were developed as a trade-off: the goal was to maximize information acquisition with the least possible submarine indiscretion. Some of these techniques began to be developed as early as World War I, when most submarine attacks were performed within the visual range, in a very close approach to

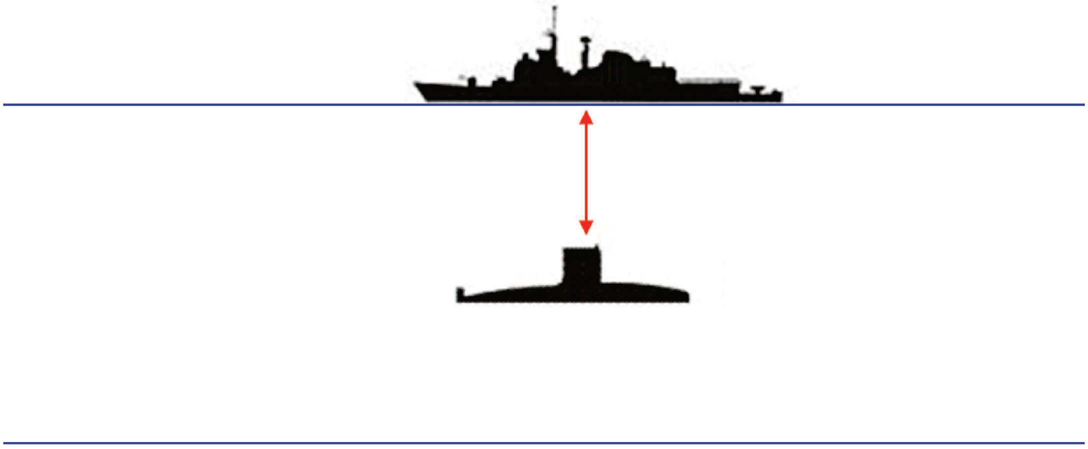


Figura 2.1: Example of safety quota

surface ships. Modern submarine weapons and sensors have ranges beyond the horizon, making it very unlikely that a submarine will get as close to surface vessels (contacts) as was typically the case until World War II. However, the skills necessary to maneuver a submarine safely in the presence of several surface contacts, using only stopwatches and mental agility, remain valuable for the training of a submariner.

The periscope is the only sensor capable of providing a complete set of data about the tactical situation around the submarine in a matter of a few seconds. However, as an optical sensor, it has all limitations naturally imposed by the light spectrum. The range of a periscope is geometrically limited at sea due to the Earth's curvature, and it is susceptible to atmospheric conditions influencing visibility. However, despite those disadvantages, periscopes continue to be of paramount importance to ensure the safety to the dived submarine, and to collect or confirm tactical information.

Due to the inherent margin of errors and uncertainty of other passive sensors, the periscope is the only sensor able to resolve with conviction the potentially questionable data sets acquired by other sensors. The operator is able to collect reliable information about the surrounding tactical situation in a matter of seconds, calculate the geographical position of the submarine and reveal the identity of targets. The information collected by a periscope usually can be used right away, because they are useful without the need of further processing (other than human interpretation)

Stanton et al. [35] presentes all challenges, risks, and strategic solutions related to submarine operations. Our work is inspired by the related issues presented in the document, where it is shown that standing at sea level breaks the submarine's invisibility and makes it vulnerable to other vessels and aircraft. Stanton et al. [35] also explains why the transition from deep to periscope depth is one of the most dangerous operations due to the potential to collide with surface vessels, our probe will allow the submarine to see at surface level from the safety quota. One of our main objectives is to minimize these risks, maximizing the surveillance operations.

2.1.2 Virtual Reality and Mixed Reality

Virtual reality involves creating an entirely digitally rendered, immersive environment on a simulation from the real world. Sensors allows a user to manipulate and move objects using controllers, head-mounted displays, and headsets.

The beginning of Virtual Reality occurred in the 50s with the invention of Sensorama [38] as can bee seen in fig. 2.2. It was a incredible technology for the time, including a stereoscopic color display, fans, odor emitters, stereo-sound system, and a motion chair.

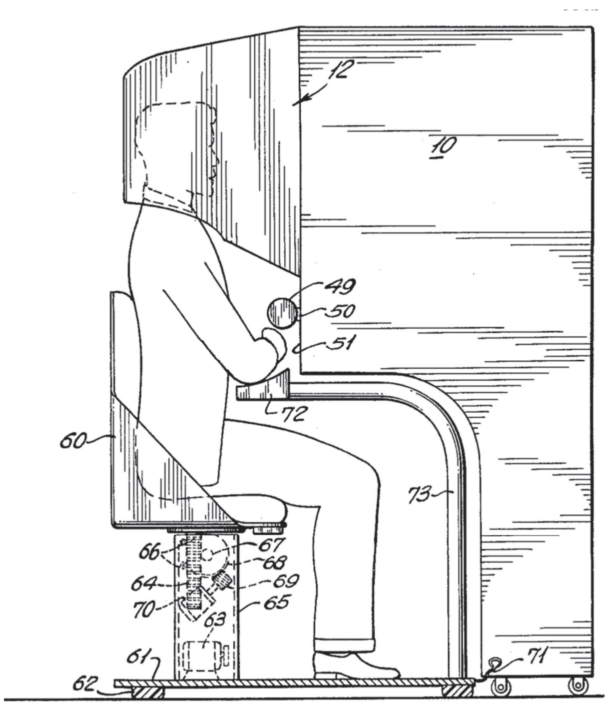


Figura 2.2: The Sensorama, from U.S. Patent 3050870

Despite being established as an emerging technology as far back, the accessibility of VR since the years 2000 has established again as a growing technology for several areas like training and storytelling.

Augmented reality refers to the overlaying of virtual content onto elements from the real world. In this scenario, the real world is central, while digital details are layered in to supplement reality.

Augmented Reality has been growing on interest from researchers and industries with its recent advances in hardware and software. Its capability to superimpose virtual models over the physical world and form a mixed reality scene is sufficient to create unique gaming and educational experiences including boat navigation experience [22].

According to [23], the Reality-Virtuality Continuum is constituted by different levels of immersion, going from the Real Environment, Augmented Reality, Augmented Virtuality and Virtual Reality. While our solution resides at the Augmented Reality stage, we use Augmented Virtuality devices as interfaces.

Mixed reality MR is a technology that uses both AR and VR approaches, creating a seamless blend between real-world elements and virtual content.

360° videos are captured with omni-directional cameras or a collection of cameras in such a way that they allow viewers to look in all directions.

Immersive technologies, consist of multi sensory digital experiences involving AR, VR, MR, and 360° videos.

2.1.3 Computer Vision and Deep Learning

Deep Neural Networks (DNN) have shown significant improvements in many applications, including computer vision. In computer vision, a specific DNN, known as Convolutional Neural Networks (CNN), has revolutionized the state of the art of object detection and recognition [34].

Most of the object detection solutions available in the literature characterize directing classifiers or locators to perform the detection.

We chose one of the most accurate fast and precise CNN available at the time, to prove our concept. Other CNN could be fit to our propose but our goal is to prove the concept. Although we did not conduced tests with others networks, since our classification results achived almost 100% of precision for the trained vessels, we considered that almost no difference would be achieved by the other.

"You Only Look Once" (YOLO) [29, 30, 31, 5] meets all the requirements for our proposal, such as the need for real-time processing and to be robust to changes in lighting. Besides that, YOLO v4 [5] is a state-of-the-art, real-time object detection system suitable for our needs. YOLO v4 is an object detector which can be trained on a single GPU with a smaller mini-batch size. Making it simpler to train the model.

The idea behind The YOLOv4 framework [5] is that a single neural network is applied to a full image. This allows YOLO to reason globally about the image when generating predictions. It is a direct development of MultiBox, but it turns MultiBox from region proposal into an object recognition method by adding a softmax layer in parallel with a box regressor and box classifier layer. It divides the image into regions and predicts bounding boxes and probabilities for each region.

The YOLO network divides the image into a $S \times S$ grid of cells, where S is a hyper-parameter defined by the user according to his needs and the characteristics of the input dataset. For each grid cell, YOLO predicts B bounding boxes for detected objects and computes C class probabilities of the objects whose centers fall inside the grid cells. The number of classes C depends on the training dataset, while B is also provided by the user.

To date, five architectures have been proposed by the authors of YOLO: the original YOLO [29], YOLO v2 [30], YOLO 9000 [30], YOLO v3 [31] and YOLO v4 [5]

Figure 2.3 shows the YOLO model, detaching its detection stage as a regression problem. On the left, we have the input image subdivided into a grid. The estimated bounding boxes (above) and the most likely class for a given cell (below) are illustrated in the center. On the right we have the bounding boxes that delimit the most likely objects detected (i.e., a dog, a bicycle, and a car).

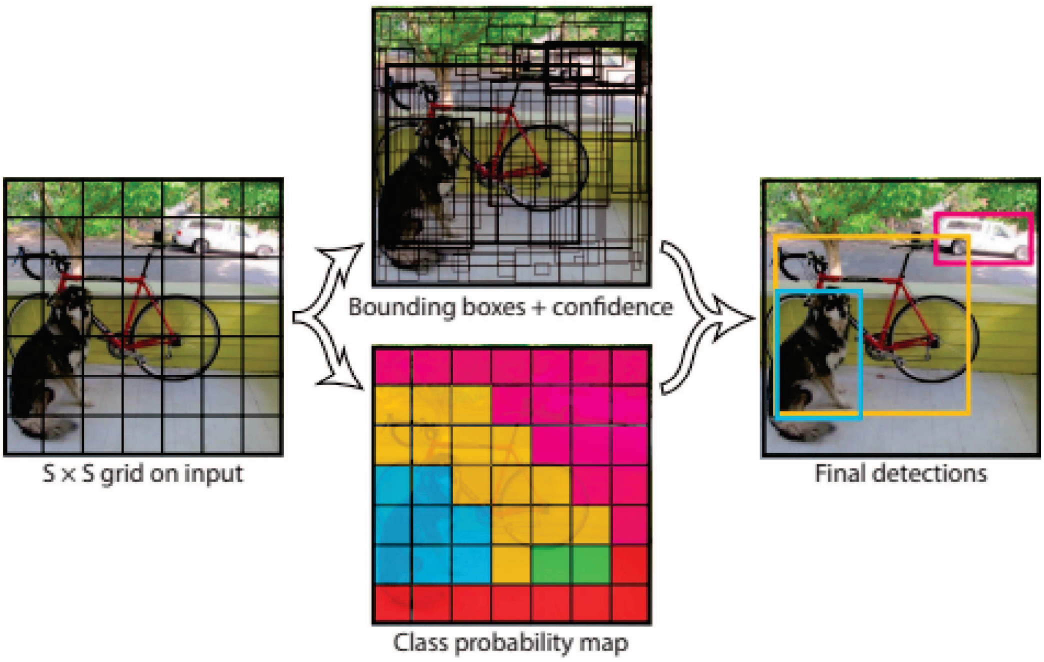


Figura 2.3: A high-level representation of the YOLO's model. Image from [29].

All YOLO versions are implemented with variations of the deep architecture called darknet [28]. The original YOLO [29] has 24 convolutional layers followed by 2 fully connected layers, the custom architecture of YOLO v2 [30] has 30 layers, while YOLO v3 [31] is a 106-layer network.

During training, YOLO uses differential weight for confidence predictions from boxes that contain object and boxes that do not contain objects, penalizing errors in small and large objects differently by predicting the square root of the bounding box width and height.

2.1.4 Fine Tuning

Fine-tuning [26] is one of the most used approach for transfer learning when working with deep learning models. It starts with a pre-trained model on the source task, usually on generic datasets, like the MS COCO dataset [21], and trains it further on the specific dataset, specializing it on the desired new task. Compared with training from scratch, fine-tuning a pre-trained convolutional neural network on a target dataset can significantly improve performance, while reducing the target labeled data requirements [39].

We used the fine-tuning strategy to train our model. For so, as showed in 2.4 we took the YOLOv4 network trained on the MS COCO dataset [21] and specialized its training on our new synthetic dataset. The first few convolutional layers learn low-level features (curves, color, edges, blobs). As we progress through the network, it learns more mid/high-level features or patterns. We freeze these low-level features trained on the MS COCO and only retrain high-level features needed for our new image classification problem, replacing the classification layer with our setting, with a different number of classes. The last few layers of the deep network can be fine-tuned while freezing the parameters of the remaining initial layers to their pre-trained values. This is driven by a combination of limited training data in the target task and the empirical evidence that initial layers learn low level features that can be directly shared across various computer vision tasks.

2.2 Related Work

Although we believe our work is the first to introduce the use of a 360° camera and Virtual Reality (VR) based periscope, other works take advantage of the combination of these technologies for surveillance and security. [12] presents the information needs and the capabilities of piloting and navigation as the paper addresses the need to assess the

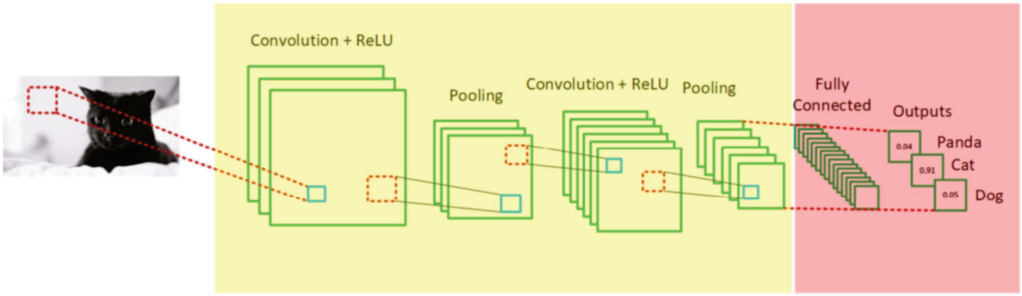


Figura 2.4: Theoretical Fine-tuning strategy, only the layers in red are trained.

impact of immersive technology on safe and effective marine transportation using Wearable, Immersive Augmented Reality (WIAR), establishing the link between technology decision support and improved navigational safety facing the problems inherent in technology introduction in marine transportation.[20] Presented an Image-based ship detection using deep learning, it uses a CNN to detect objects and then classify as ship, speedboat, and buoy.

[18] presents a systematic review analyzing the publication type, the AR device, which information elements are visualized and how the information is displayed, based on the information, we displayed on the XR device basic information, like own ship speed and relative heading.

Detecting moving objects in video streaming is essential for complex applications, such as object tracking and video retrieval. Moving object detection finds elements in the foreground in order to extract helpful information from the environment.

The literature on XR periscope or Computer Vision applied for submarines' periscope is almost nonexistent. Still, this problem faces similar issues with detecting cars or traffic signs using a camera in autonomous or semi-autonomous vehicles. [10] has proposed a method to generate artificial traffic-related training data for deep traffic light detectors, offering a solution using deep neural networks for problems associated with autonomous driving. Concerning vessel detection and classification, [16] proposed a novel probabilistic ship detection and classification system based on deep learning using a dataset of images available at the web. However, the annotation data from different classes of ships are not vast and not trivial to be solved, as will be seen further as we used a synthetic dataset the annotation was made in a semi-automatic way, and in the same way we produced images from all angles of each ship.

Availability of domain-specific datasets is an essential problem in object detection. Datasets of inshore and offshore maritime vessels are no exception, with a limited number of studies addressing maritime vessel detection on such datasets. Ship detection in a traditional setting depends extensively on human monitoring, which is highly expensive and unproductive. Moreover, the complexity of the maritime environment makes it difficult for humans to focus on video footage for prolonged periods of time [33].

In this work, we intend to detect vessels in images using Computer Vision techniques. In order to create a dataset, we used the Bridge Navy Simulator for producing a set of renderings of strategic ship classes for submarines operations, in Fig 2.5 we can see a overview of the Bridge simulator. Similarly, [36] proposed a synthetic dataset to classify ships from satellite images. We took a similar approach developing a dataset composed of synthetic images with a different camera position, constrained to the submarine periscope point of view. Maritime vessel detection from satellite was employed in many studies, a review from 2018, has gathered a large number of papers about classification from satellites images [15].

Figura 2.5: Navy Bridge Simulator, used for extracting synthetic images for the proposed classifier.

As discussed in [27], transfer learning techniques reduce the need for large datasets due to the generalization ability of the parameters learned by the lower layers of the CNN from public datasets, like MS COCO [21]. We used a pre-trained YOLOv4 [5] CNN to get such parameters and train the weights of the classification layers with our synthetic dataset.

Capítulo 3

An Periscope for Submarines with Extended Visual Classification

Section 3.1 describes our novel periscope architecture, and the following sections describes how it is implemented, detailing our proposed classification strategy, our training dataset, our object distance estimation approach and our developed software functionalities.

3.1 Proposed Solution

We propose a novel generation of submarine periscopes based on a high resolution 360° camera mounted in a floatable probe, coupled to a Virtual Reality HMD device. The probe is projected in such a way that it can be dragged by an underwater vehicle (submarine). It has a precise hydrodynamic to achieve stability in the camera image and enough height to extend the horizon line and detect surface elements and vessels. The 360° video is streamed to the HMD device, placed inside the submarine. The movement of the HMD performs the selection of the 360° video area being viewed by the periscope operator and processed by the Computer Vision module. AR features are inserted in the image, including vessel type, bearing, and distance calculation information.

The submarine velocity at deep waters is around 5 knots. In this sense, the probe was developed in such a way that it has stability and hydrodynamics at this speed. In order to avoid wave and water drops interference in the images, the camera was projected to be mounted at 40 centimeters above sea level. The camera is attached to a protected HMDI cable that connects the devices with the submarine. Fig.3.1 shows the schematic view of our solution, and Fig. 3.2 shows our operational-developed probe, with the camera

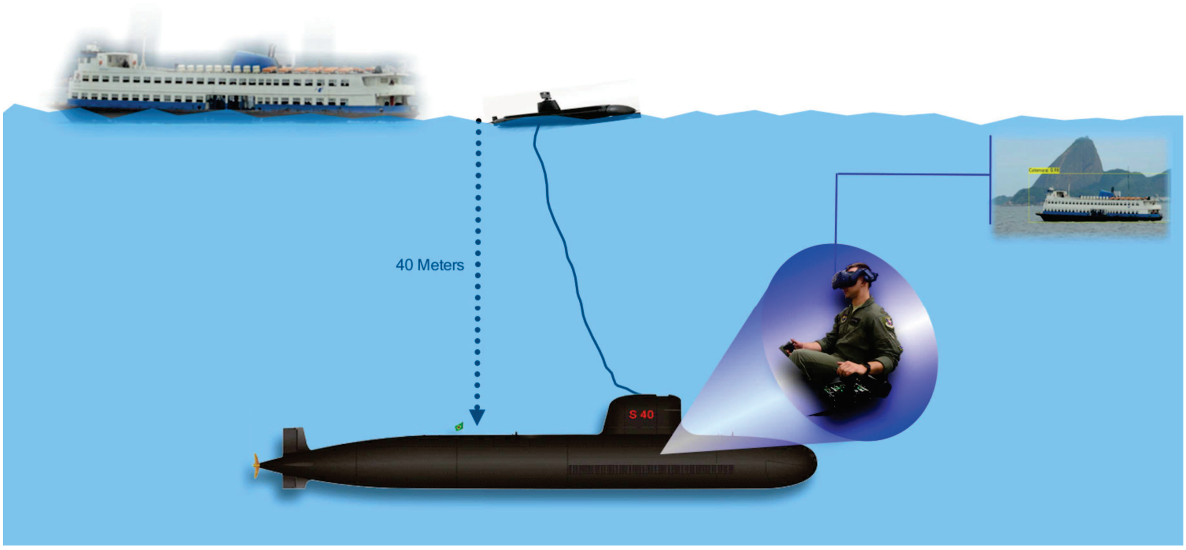


Figura 3.1: Overview of the proposed solution.

mounted at the top, and Fig. 3.3 shows the probe being placed in water for in field test.

Figura 3.2: Probe Mockup with a 360° camera.

The targeted areas of interest are processed by deep learning algorithms using CNN for feature classifications (Section 3.2). The CNN is trained with a dataset composed of 99,000 synthetic images (Section 3.3) of ships generated using the Brazilian Navy Bridge Simulator, from the Naval Systems Analysis Center (CASNAV). We made the generated dataset open access and collaborative, suitable for future extensions.Our dataset is open access, collaborative and can be accessed at [1]. Although we have built our system for the set of vessels considered most important for the Brazilian submarine operations, it is straightforward to expand and include more vessel models in this dataset. The distance to each target is estimated by the relationship between the actual known height of the detected and classified vessels and the vessel's size in the recorded image (Section 3.4).

Figura 3.3: Probe Mockup on water.

3.2 Classification Stage

Once the images are transmitted to the submarine, we apply different Computer Vision approaches for enhancing and detecting features above the sea, based on the YOLOv4 framework [5], which is explained in details on section 2.1.3.

After the training stage, as we input an image, the CNN returns several axis-aligned bounding boxes. Each bounding box is defined by (x, y), w, and h, where (x, y) is the center of the box, and w and h are its width and height, respectively. By multiplying the conditional class probability and the individual box confidence predictions, we get the class-specific confidence score for each box and use this data to draw the boxes on the output image. The height of the box and additional information about the ship's class are used to calculate the object distance.

3.3 Training Data

Due to the periscope's positioning and our 360° camera elevation above water, it is plausible to state that the objects on the surface necessarily cross the horizon line. All of our training data was generated with this concept in mind, and the virtual camera used in the simulated scenario positioned about 40 centimeters above water level. Fig. 3.4 illustrates this point of view and configuration.

We developed an application in order to extract synthetic images of five classes of ships already implemented from the CASNAV simulation system. Samples of generated images for each ship class can be seen in Fig. 3.5 to Fig. 3.9. For each ship class, we generated

Figura 3.4: Periscope point of view, adapted from [24]

one image for each degree step in a bow angle (from 0° to 359°). This set was combined with different backgrounds and distance conditions, as described in Table 3.1, leading to 3,960 images per class and 19,800 in total. For each "closest" positioning distance, we generated the image with and without background, so our CNN network could learn to detect small details available at each vessel. Images without background means that besides the ship, the image also contains water and sky. Image with background means that it also has land behind the vessel. Although our database contains only 5 types of vessels, our system allows to easily extend it with other categories.

The synthetic dataset including the above-mentioned 19,800 images is quite repetitive, since we assume 1° steps in bow angle and too clean renderings (i.e., without noise). Doing so, we noticed that our results presented large overfitting rates. To avoid this, we included in our dataset new images generated through data augmentation strategies. We found that the following types of data augmentation were the most relevant in our dataset: Gaussian noise, impulsive noise, blur, shadow, shear, and small rotations restricted to angles that can be included by sea waves movement.

We generated four augmented images for each synthetic image in the initial collection of renderings, assuming random values defined between a minimum and a maximum parameter for each original image. By doing so, we end up with a synthetic image dataset composed of 99,000 images. Fig. 3.10 shows an example of an augmented synthetic image.

ShipType	**Bg**	**Distance (in meters)**									
		500	1,000	2,000	3,000	4,000	5,000	6,000	7,000	8,000	10,000
ContainerShip	Yes		✓	✓		✓		✓		✓	
	No		✓	✓		✓		✓		✓	✓
Ferry	Yes	✓	✓	✓	✓						
	No	✓	✓	✓	✓	✓		✓		✓	
Frigate	Yes	✓	✓	✓	✓						
	No	✓	✓	✓	✓	✓		✓		✓	
PassengerShip	Yes		✓	✓		✓		✓			
	No		✓	✓		✓		✓	✓	✓	✓
YardShip	Yes	✓	✓	✓	✓						
	No	✓	✓	✓	✓	✓	✓	✓			

Tabela 3.1: Artificial images generated for fine-tuning the CNN. We consider five classes (Ship Type), the presence or the absence of background (Bg), and different distances of the object to the camera.

Figura 3.5: Container Ship

In order to fine-tune our CNN model, it is necessary to have all the data with precise annotation. Due to the large number of images, it was impossible to label them one by one manually, so we implemented an approach for tagging them in a semi-automatic way. The script was developed in AutoIt [2] and after the user inputs the position of each ship in each distance at 90°, 60°, and 30°, it calculates and generates a file for each image in the YOLO's annotation format:

```
class x y width height
```

Figura 3.6: Ferry

Figura 3.7: Frigate

3.3.1 Vessel Classification Software

In order to make possible a large and incremental image database , we proposed a pipeline composed by a classification Vessel system. This system has a test-bed user interface with additional functionalities and composed by two modules: a Data Acquisition Module (DAQ) and a Graphical User Interface (GUI).

Data Acquisition Module (DAQ) This module consists of a set of Python scripts for CNN calculating and GPU processing using OpenCV [25], an open-source Computer Vision and Machine Learning software library. This module assists in building an image

Figura 3.8: Passenger Ship

Figura 3.9: Yard Ship

dataset of real vessel images.

Graphical User Interface (GUI): This module was developed using AutoIt [2] for the GUI, whose control panel is presented in Fig. 3.11. With the GUI module, the user can choose parameter values like the maximum frame rate for video sequence acquisition, the camera used (with previously calculated focal length f) and filters in the displayed results. With this interface, the user can also enable the DAQ module to capture images to populate a dataset with real vessel images. When the DAQ is enabled, the system loads the public COCO trained weights on the CNN model, filtering detection for only reporting objects of the *boat* class. Once the user annotates the ship type that he/she

Figura 3.10: Example of augmented synthetic image used for training the classification module. This image includes Gaussian noise and blur.

is observing, the system starts to save the video frames at the frame rate chosen by the user, and includes the location of the bounding box given by the CNN and the ship class in the respective annotation file. The idea of this procedure is to acquire real data for further improvement of the trained detection and classification model. The communication between the GUI module and Python modules are through console in/out comunication

3.4 Distance Estimation Stage

The optical periscope calculates the distance of a known object by a stadimetric rangefinding method, which is a process based on triangulation in which the angle subtended by a target of known height (usually waterline to masthead height) is measured by vertically displacing the fields of view in each half of a split lens. This optically measured angle and the operator-inserted target height are used to estimate the distance to the target in yards (a.k.a. target range). A typical stadiometer split image can be seen in Fig. 3.12.

The formula to manually compute the target range (TR) is:

$$TR = \frac{TargetHeight \times FocalLength}{StadimeterSplit}. \tag{3.1}$$

The distance from objects is the most important calculation when the submarine is at periscope depth and detects a vessel. The faster and precise the distance is calculated,

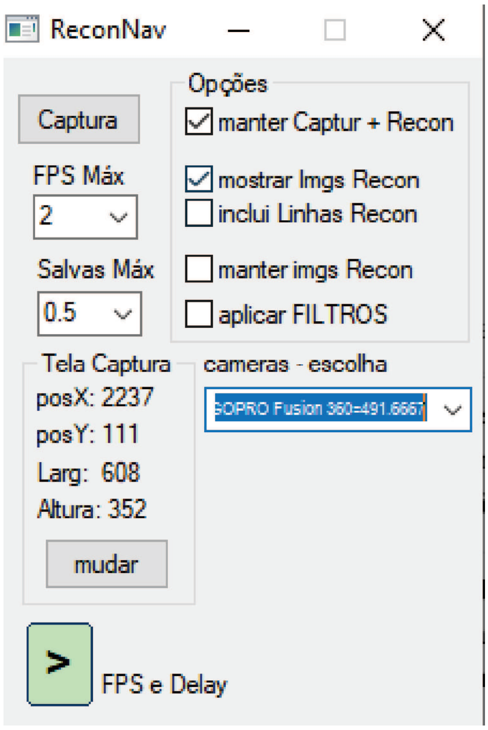


Figura 3.11: The control panel of the Graphics User Interface (GUI) of our system.

the less time the periscope has to be hoisted. We have developed a stadimeter-inspired method for estimating the distance of ships of known classes through the classified image results. Our approach is based on triangle similarity, where three parameters are necessary to calculate the ship distance:

1. **Height of the object in the 3-dimensional space (H)**: Once we have classified the vessel, it is possible to retrieve its known height since this information is usually available to the periscope officer.

2. **Height of the object in the image (P)**: After the application of the detection and classification model, we get as a result the axis-aligned bounding box of each detected ship and its respective confidence score indicating how good the detection is. We assume that the height of the bounding box is the height P of the object in image space, measured in pixels.

3. **Camera focal length (f)**: It can be found in the camera's specifications or estimated using one of the methods explained below. As depicted in Fig. 3.13, it can be computed as:

$$f = \frac{W}{2} \times \cot\left(\frac{\alpha}{2}\right), \tag{3.2}$$

where α is the field of view angle, and W is the image width. Our approach for

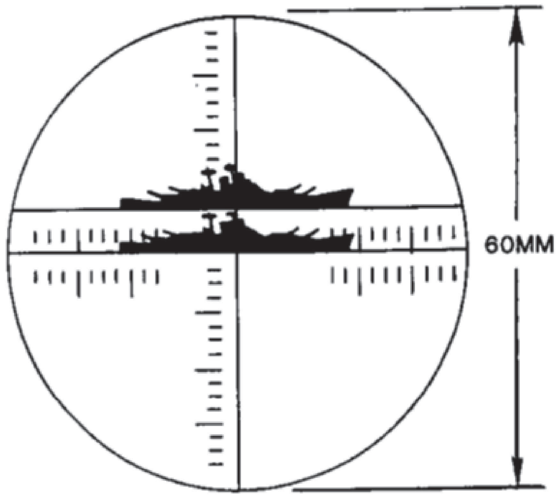


Figura 3.12: Typical stadiometer split image, The image was extracted from the periscope manual [17].

computing the focal length is:

$$f = \frac{P \times D}{H}, \tag{3.3}$$

where D is a known distance of an object used for calibration, P is the height (in pixels) of the object in the image, and H is the known height of the object's class in the 3-dimensional space.

As mentioned in Section 3.2, the bounding box of each detected object is defined by (x, y), w, and h, where (x, y) is the location of center of the bounding box, and w and h are its width and height, in pixels. As the heights of the trained classes are known, after the model returns a bounding box with an appropriate confidence level, it becomes

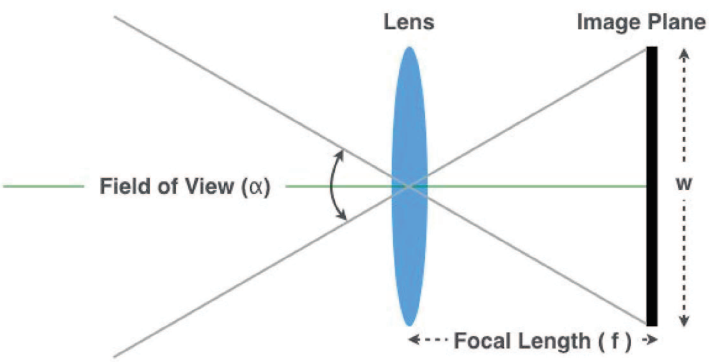


Figura 3.13: Camera geometry.

possible to calculate the distance of a target object to the submarine using:

$$D = \frac{f \times H}{P}, \tag{3.4}$$

where $P = h$ by construction.

Capítulo 4

Experiments and Results

Figure 4.1 shows a flow chart of the steps. The first stage is called Database Generation and is composed by the simulator data generation and acquisition, followed by the data augmentation, training and data set validation. The second stage is named as Model Configuration and Training and starts with the image labeling process through our developed system, the Yolo framework execution and training the data into the cloud environment. Finally, the third stage is the final user process, composed by the real time CNN classification and Data Acquisition for enhancing the classified data with new images.

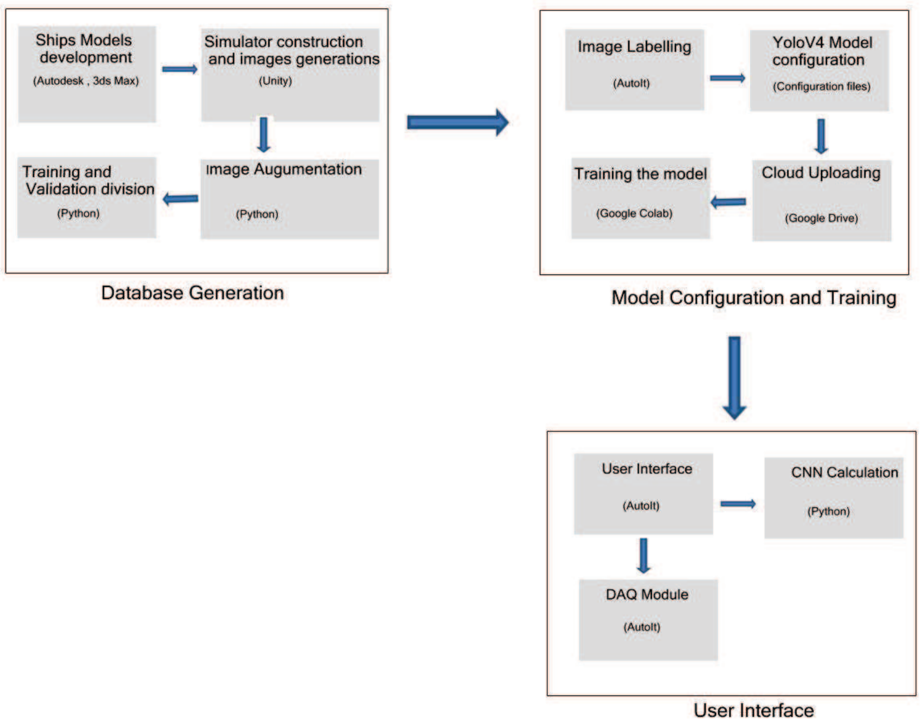


Figura 4.1: Steps Flow Chart

The following subsections describe the chosen tools for the system development, how we have implemented the proposed system, trained our detection and classification model, evaluated our results using synthetic and natural images, and performed tests simulating real conditions. Our Solution is summarized by the following steps:

1. **3D Ship Modeling**: Ship Models were developed with Autodesk and 3DS Max;
2. **Images generation**: The images were extracted from our Bridge Simulator, that is built in Unity;
3. **Augmentation and dataset division** : A Python script was developed to perform image augmentation and dataset division in training and validation datasets;
4. **Image labelling**: An AutoIt script was developed to label the images;
5. **Model training**: The configuration files and images were uploaded in the cloud, for training the model through Google Colab;
6. **CNN and user interface**: The CNN classification is performed;.
7. **Data Acquisition (DAQ)**: The DAQ module was developed in AutoIt.

4.1 Chosen Tools

In order to implement and test our solution we choose some tools to make a working prototype, In the next subsections we appoint the software and hardware used in our implementation.

4.1.1 YOLO V4

As mentioned in section 2.1.3, for the vessel classifications we choose the YOLO architecture.

The input image is divided into an S x S grid. If the center of an object falls into this grid cell,that cell is responsible for detecting that object. Each grid predicts a number of bounding boxes and confidence scores for those boxes. Confidence here is defined as Probability of an Object multiplied by the thresholded IoU score, where IoU scores that are less than 0.5 mean that the confidence is close to zero.The bounding box is defined by x, y, w, h where x, y are the center of the box and w and h are the height and width, By multiplying the conditional class probability and the individual box confidence predictions, it is possible to get the class-specific confidence score for each box

4.1.2 Python

Python is a high-level, interpreted, scripted, imperative, object-oriented, functional, dynamic typing, and strong programming language. It currently has an open source community development model managed by the nonprofit Python Software Foundation. In our proposal we adopted Phyton for processing the input images.

4.1.3 OpenCV with CUDA

We adopted OpenCV for image processings tasks. The OpenCV DNN module supports deep learning inference on images and videos. It does not support fine-tuning and training. OpenCV (Open Source Computer Vision Library) [25] originally developed by Intel in 2000, is a multiplatform library, completely free for academic and commercial use for application development in the fields of Computer Vision and image analysis.

Modern GPU accelerators has become powerful and featured enough to be capable to perform general purpose computations (GPGPU). Significant part of Computer Vision is image processing, the area that graphics accelerators were originally designed for. Other parts also suppose massive parallel computations and often naturally map to GPU architectures. So it's challenging but very rewarding to implement all these advantages and accelerate OpenCV on graphics processors.

4.1.4 CASNAV Simulator

The simulator was build on UNITY. The Ship models were developed in Blender and Autodesk 3ds Max. The simulator was used for capturing the synthetic images of the vessels, allowing a complete control of the virtual camera positioning, background conditions and lighting variations.

4.1.5 GoPro Fusion

For purpose of this study and conception test, we used a Gopro Fusion 360 camera for data aquisition, installed at the top of our built probe.

The GoPro Fusion is a spherical camera that can capture video at up to 5.2K/30p or 3K/60p. Since it "over-captures"it is possible to turn spherical content into traditional stills and videos. It also features with an advanced "gimbal-like"image stabilization system.

4.2 Model Configuration and Training

As described in Section 3.3, the dataset was generated using synthetic data from the CASNAV Bridge simulator and extended through the data augmentation approaches. Before training the model, the dataset was randomly divided in a *training* dataset with 79,200 images and a *validation* dataset, composed of 19,800 images.

The model was trained using the publicly available Darknet, which is an open-source neural network framework written in `C` and `CUDA`. It includes the implementation of a consolidated state-of-the-art object detector, YOLOv4.

In order to improve the dataset labeling process, we developed a program that semi-automatize it. Fig 4.2 shows an example of labeled image produced by our program.

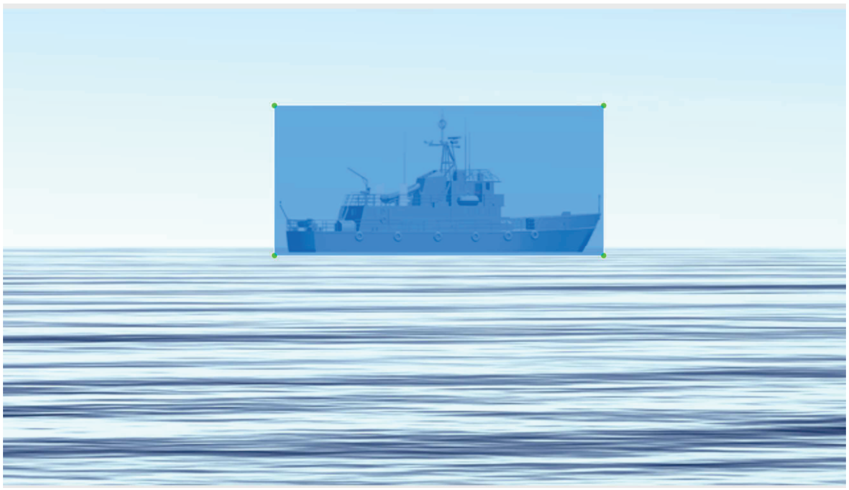


Figura 4.2: Labeled Image

We have used default values for almost all YOLOv4 hyperparameters. The only exceptions are: the input image size, which was set to 640×352 pixels; the batch size was set to 64 and subdivision to 16; the size of the last convolutional filters before each of the YOLO layers was set to 30, which is the result of $\texttt{classes} + 5 \times 3$ [1], as it depends on the number of classes according to Darknet documentation. The model was retrained

[1] Where 5 and 3 are constants.

using 70K iterations, keeping the weights for every 10K iterations. The number of steps was set to 56K following the recommendation of 80% for the number of batches for this hyperparameter.

The other necessary files are:

- obj.data: contains information like the number of classes and the path of others configurations files;
- obj.names: contains the name of the classes on the correct order;
- train.txt and test.txt: hold the relative paths to all our training images and validation images.

After all the configuration files were prepared and the dataset labeled the model was trained and the fine-tuning strategy was applied, so the weights for the convolutional layers (yolov4.conv.137) of the YOLOv4 network trained on the coco dataset was downloaded. By using these weights, it helps the custom object detector to be more accurate and not having to train as long.

All training and interference processes were performed on Google Colab Professional [4], which is a research project for prototyping machine learning models on powerful hardware options such as GPUs and TPUs. It provides a serverless Jupyter notebook environment for interactive development. The hardware used to train our model was: Intel(R) Xeon(R) CPU @ 2.00GHz, 26GB RAM, 200GB Hard Drive and a NVIDIA Tesla V100-SXM2-16GB GPU.

4.3 Data Acquisition Interface

In order to improve the labeling process, we developed a dedicated tool, as shown in fig. 4.3. Its main objective is to allow the visualization of a video in real time with recognition boxes around the objects defined at each frame. These boxes have on their top the type of recognized object (in this case the ship type) and its distance in meters. A second function of the tool is to allow the images with the boxes in the saved objects to be stored for logging or event recording purposes. Finally, this program also intends to store the original images and their reconnaissance parameters, so users can include and train new vessels in the future.

The system saves the object type, its position, length, height and distance in compatible format for injection into the YOLO v4 system to enhance model training.

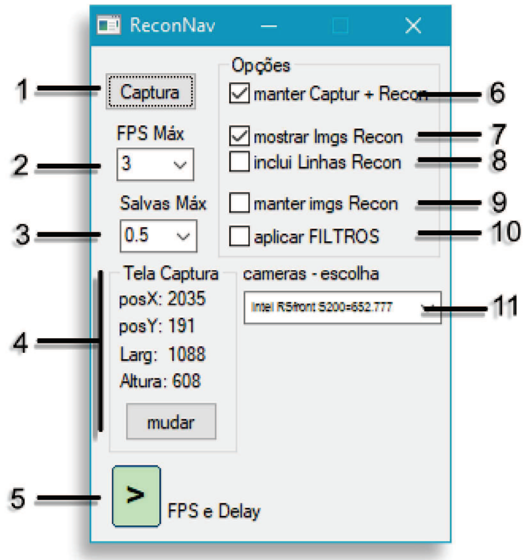


Figura 4.3: System for helping vessels classification

Our tool has the following funcionalities:

1. **Capture**: Defines the folder used as the workspace and storage of the captured images;

2. **Maximum FPS**: Maximum amount of frames to be captured from the video, which will be delivered to the pyhton module for recognition. The actual amount captured may be less if the recognition speed is less than the FPS.

3. **Max Saves**: Defines how many images per second will be saved when the options to keep captured images for further training (third function of the program, activated by option 6) and/or keep Recognized images for log or record (second program function, option 9) is ON. This rate will be less than or equal to the actual captured frame rate.

4. **Screen Capture**: Defines the location of the screen where the capture will be made and its size.

5. **PLAY**: Activates capture, possible image display with recognition and any selected saves

6. **Keep Capture and Recon**: Informs that the third function of the program must be activated, that is, save the images and their recognition parameters for further training. The frame rate per second is set in option 3.

7. **Show Recon Images** : Plots the images that have recognition objects .

8. **Include lines on the Label Files**: Include the lines with parameters of the bonding box on the label files.

9. **Keep Recognition Images**: Keep on the hard drive the images that has detected objects

10. **Filters**: Apply filters on the images before CNN recognition

11. **Cameras**: Select a predefined camera described in the configuration file, with its focal length parameters.

4.4 Classification Results

The model was tested either with the validation dataset and with real vessel images, always restricted to the point of view of the periscope. Following we describe in details each step.

4.4.1 Detection and Classification

For measuring the results we used the Mean Average Precision (mAP) approach. After the model training, we noticed a fast convergence to an optimal average loss, as can be seen in Fig. 4.4.

The mAP results stabilized when using 10K iterations or more. After running the model with different weights on real data previously acquired, it was empirically defined that the best achieved result was at 20K iterations.

Ship Type	**mAP**	**TruePositive**	**FalsePositive**
Container Ship	99.96%	4,877	4
Ferry	99.99%	4,973	2
Frigate	99.99%	4,988	8
Passenger Ship	100.00%	255	0
Yard Ship	99.98%	4,877	10

Tabela 4.1: Results of Mean Average Precision (mAP).

The precision results can be checked in Table 4.1. The global model results are: Precision = 1.00, Recall = 1.00, F1-score = 1.00, True Positive = 19,787, False Positive = 24, False Negative = 13, and Average IoU (Intersection over Union) = 0.90.

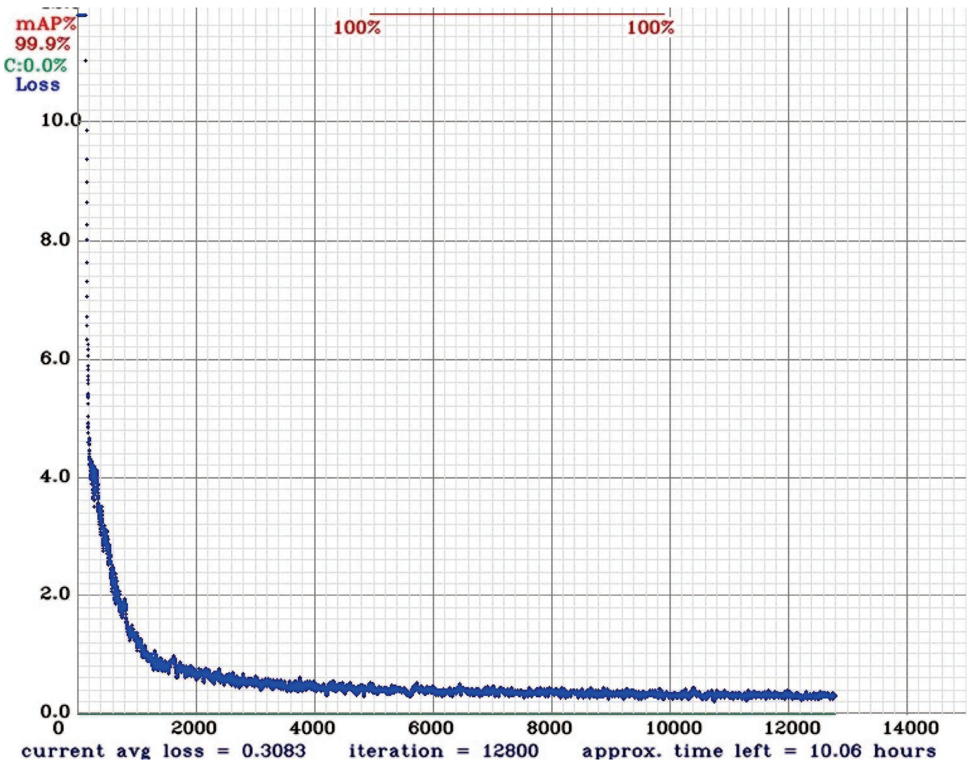


Figura 4.4: Average loss at 12K iterations.

We believe that this good mAP result is related to the similarity of training and validation datasets and because the best weights were at 20K interactions. It is well-known that with the increase of iterations, the model might present overfitting. However, it is important to remember that the operational CNN will be applied to real images and not to synthetic ones, which are very different from the training dataset and not prone to this overfitting issue.However, The confusion matrix can be seen at Figure 4.5 and confirms the information from Map analysis.

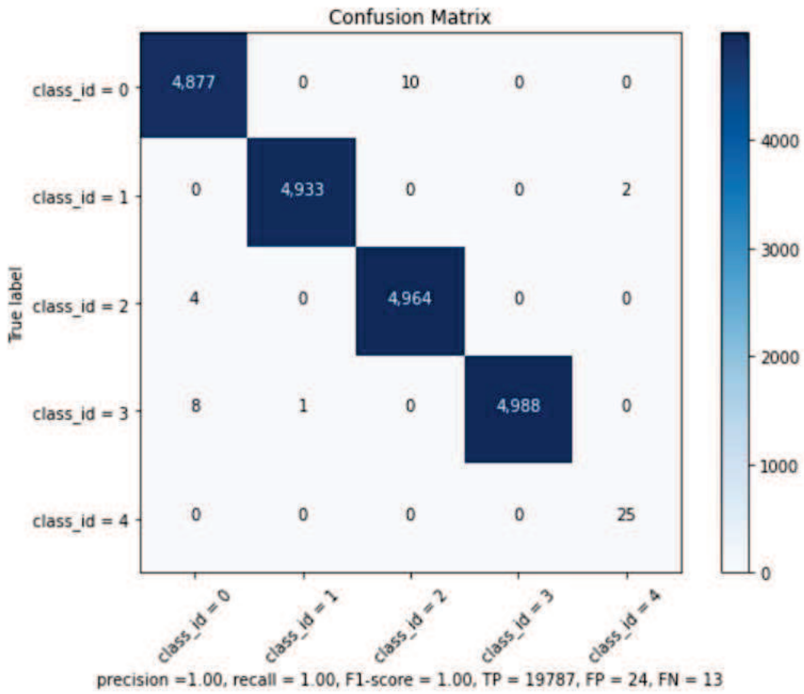


Figura 4.5: Confusion Matrix.

Figures 4.6 to 4.10 show examples of the model applied to synthetic images from the validation dataset. As theoretically predicted in the mAP analysis, it is possible to see that the model achieves a very good grade of precision and classification even with greater distance and different bow angles.

Figura 4.6: Container Ship, Distance = 2000 Yards, Bow Angle = 15

Figura 4.7: Ferry, Distance = 4000 Yards, Bow Angle = 160

Figura 4.8: Frigate, Distance = 2000 Yards, Bow Angle = 210

Real world image testing We also tested our model on real images of the same kind of vessels and achieved good results, as can be seen in Figures 4.11 to 4.15.The vessels were detected in all the images, and the most likely class associated with each of the detections are to the correct class of ship. As can be seen in Fig. . 4.13, our solution detected both Frigate and Yard Ship, due to similarities on both class of vessels.

Figura 4.9: Passenger Ship, Distance = 4000 Yards, Bow Angle = 332

Figura 4.10: Yard Ship, Distance = 4000 Yards, Bow Angle = 332

4.4.2 Field Testing

Finally, we tested our classification model using images captured from our XR periscope device. The probe uses a GoPro Fusion camera that captures video at up to 5.2K/30p or 3K/60p. Since it over-captures, it is possible to convert spherical content into traditional stills images and videos sequences. This camera has an advanced "gimbal-like" image stabilization system that prevents the inclusion of movement artifacts in the

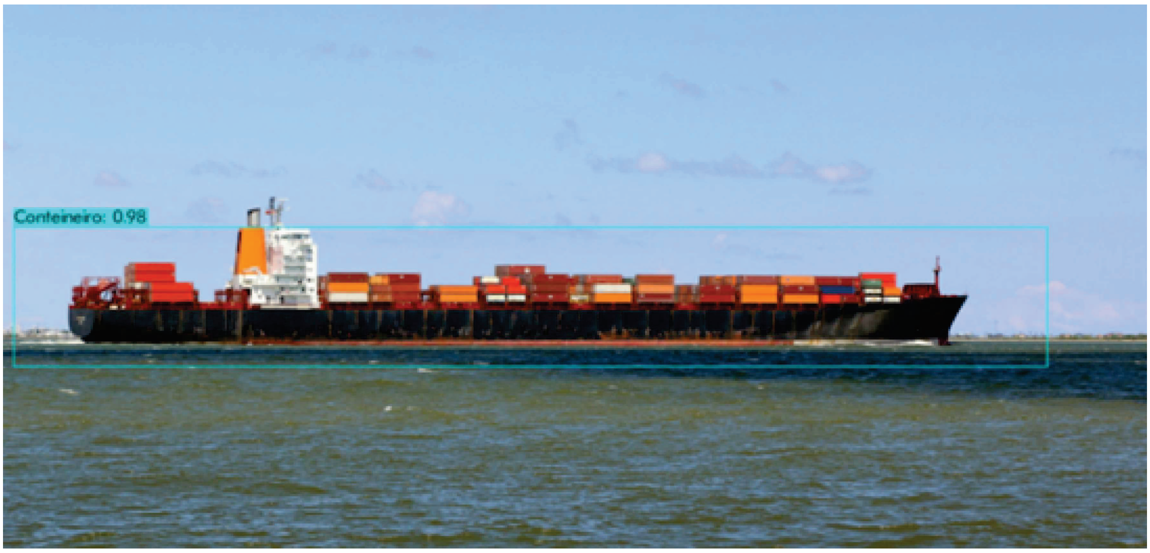


Figura 4.11: Container Ship

Figura 4.12: Ferry

captured images.

We tested our solution, including software, equipment, and probe, with a Yard Ship of the Brazilian navy. Fig. 4.16 and Fig. 4.17 illustrates the detection, classification and distance estimation of the target ship in two frames of the video sequence. Note that results are consistent even under challenging weather conditions, with poor natural lighting due to a cloudy/raining day and image distortions resulting from the camera lenses being often wet with saltwater.

Figura 4.13: Frigate

Figura 4.14: Passenger Ship

Figura 4.15: Yard Ship

It is possible to see a Yard Ship at 15.38 and 62.69 meters, respectively. This test was performed using frames of a video captured by our probe

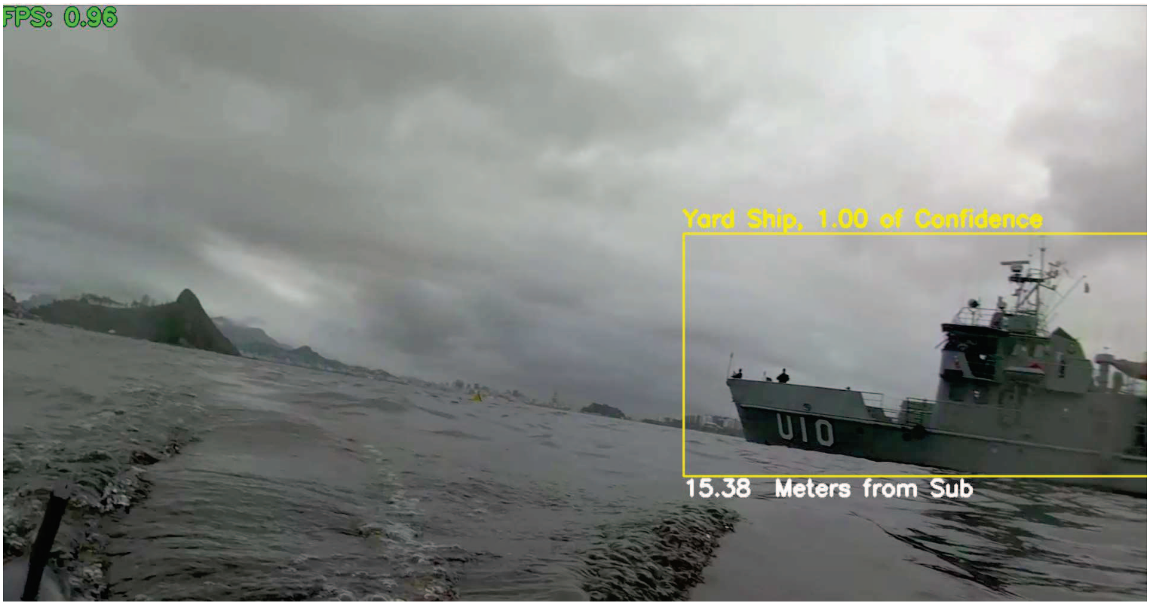


Figura 4.16: Yard Ship at 15.38 meters

Figura 4.17: Yard Ship at 62.69 meters

4.5 XR Periscope User Experience

In order to validate our XR periscope proposal as a whole, we developed an experimental scenario and procedures. Although the submarine community is not very large, with the help of the Brazilian navy we were able to perform a simulation and questionnaires with 19 experienced submarine officers. The tests were conducted at the Brazilian center for submariner training, in the same simulator where the officers are training as periscope officers. All the participants are males, submariners with ages from 25 to 54. A preliminary test was developed with 2 officers to calibrate the procedure.

The procedures and results are described in the following sections.

The experiment consists in a scenario where the submarine officer is told that he has to perform a horizontal scan procedure with the XR Periscope. We formulated the following hypothesis in order to validate our proposal:

- Hypothesis 1 - The XR Periscope improves the security in the procedure to return to periscope depth;
- Hypothesis 2 - The submarine tasks that involve observation of points and vessels of interest can be performed from the security quota with the XR Periscope;
- Hypothesis 3 - The ship recognition, classification and distance estimation improves the navigation process;

- Hypothesis 4 - The XR Periscope contributes to lower the general Submarine Discretion Fee (SDF).

Since there isn't yet a tactical and safety procedure defined by the Navy to use the equipment, we were not allowed to conduce the experiment using a real submarine. For this reason, we recorded a set of videos with the probe and used the 360 videos with the VR device, simulating the environment of a real periscope operation. We decided that for this test proposal the duration of the video was 45 seconds, similar as the maneuver with the optical periscope, which has to be 30 seconds according to the perisher technique. Once the image recognition procedure was performed, 90 sequential frames were saved to be used experimentally with the virtual reality glasses, resulting on a 2 FPS rate.

In order to visualize and manage these images in virtual reality, the Unity platform [14] was used. In order to give the freedom of the head movement, we developed a simple scenario composed of a sphere and the 360 video projected on it through a skybox.

We implemented this visualization solution using a sphere with inverted normals and projected the video sequences into it, as shown in 4.18.

In this way, we positioned the main camera of the project (which will be the user's head in the VR glasses) in the center of the sphere, giving the impression of being in a 360° environment determined by the selected image.

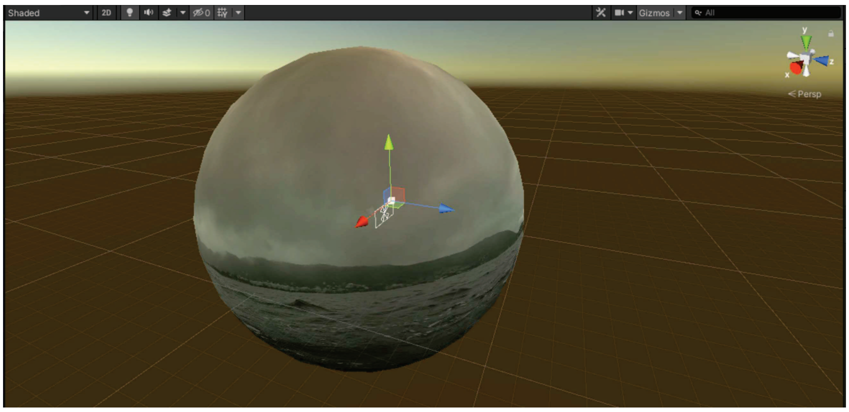


Figura 4.18: Skybox

To perform the scene change, a simple algorithm was created that moves to the next photo every "X"seconds, changing the original 2D texture of the inverted material mentioned above, through a list of 360° images generated by the algorithm, thus giving a sense of continuity in the scene and extremely lightly simulating a sense of movement.

The relative direction of the user's head in relation to the front of the vessel is calculated, which is displayed in the simulation along with its heading and speed at each instant. To perform this calculation we use the y position of the camera in relation to a fixed offset determined by the photo that indicates the position of the front of the vessel in relation to the original point of rotation of the photo image. This value can be recalculated with the variation of the images and undergo minor changes.The (simulated) submarine heading and speed are also displayed.

A radial light point was included with the camera in the center of the sphere, so that the image brightness could be controlled evenly at all points, adjusting according to the best view of the scene. Thus, when running the simulation, we have a sequence of images treated in 360° around the user, showing how they would be visualized through the use of image recognition and augmented reality.

In order to reproduce a real situation, we created a fictional scenario, using the most likeable phraseology and procedures of a submariner as possible. All participants have large experience with periscope operation. Each officer was told as the simulation began that

The submarine is in a fictitious location; the commander informed the periscope officer that he should perform a horizon scan from a security quota. The commander informs the periscope officer that he must use a new system that is in the final stages of development by CASNAV and UFF University, the XR Periscope. The Procedure developed by COMFORS indicates that the system must be hoisted to the surface from a depth of 42 meters as in fig. 3.1 for a time of 45 seconds with a maximum speed of 5 knots.

Figure 4.19 shows the user visualization on the XR device with the heading, speed and relative bearing information in blue. Also, the recognized objects are highlighted with its distance estimation.

After the procedure, each officer filled a survey, considering his experience using a Likert scale [37]. Questions 1 asks the level of experience of the user with the horizon scan procedure, and question 2 the experience with virtual reality. As can be seen in fig. 4.20 the users have a good experience with the optical periscope horizon scan and median level of experience with virtual reality.

Questions 3, 4 and 6 are related to our hypothesis 1 and 2:

- Question 3 - Does the XR Periscope helps in compiling the tactical scenario?;
- Question 4 - Assuming the XR periscope can be launched from the security quota,

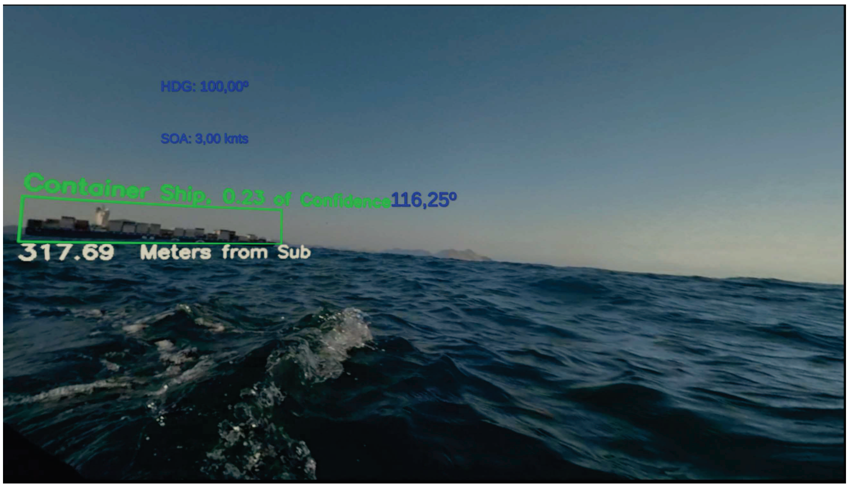
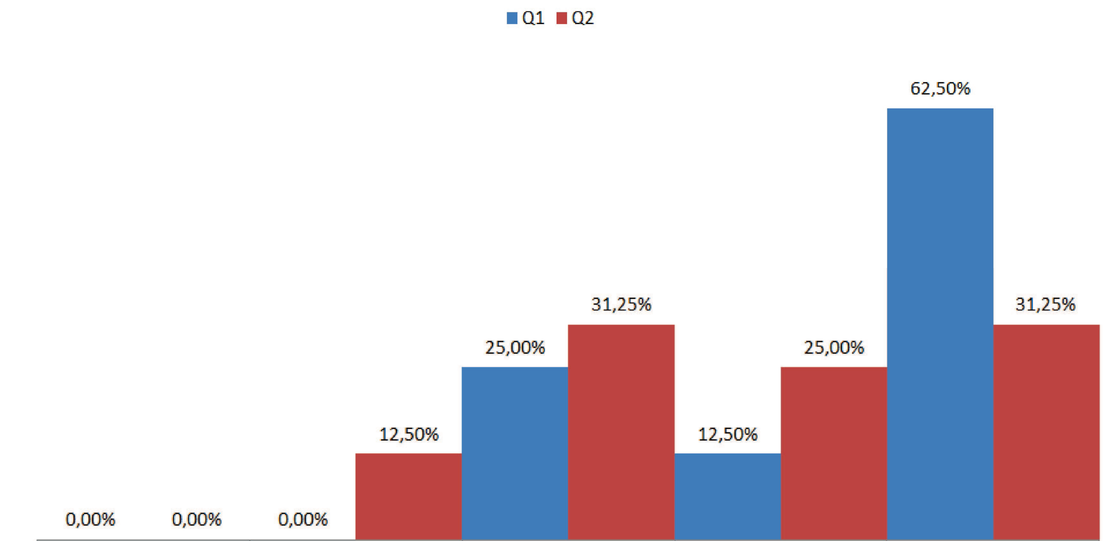


Figura 4.19: User Interface of the XR device



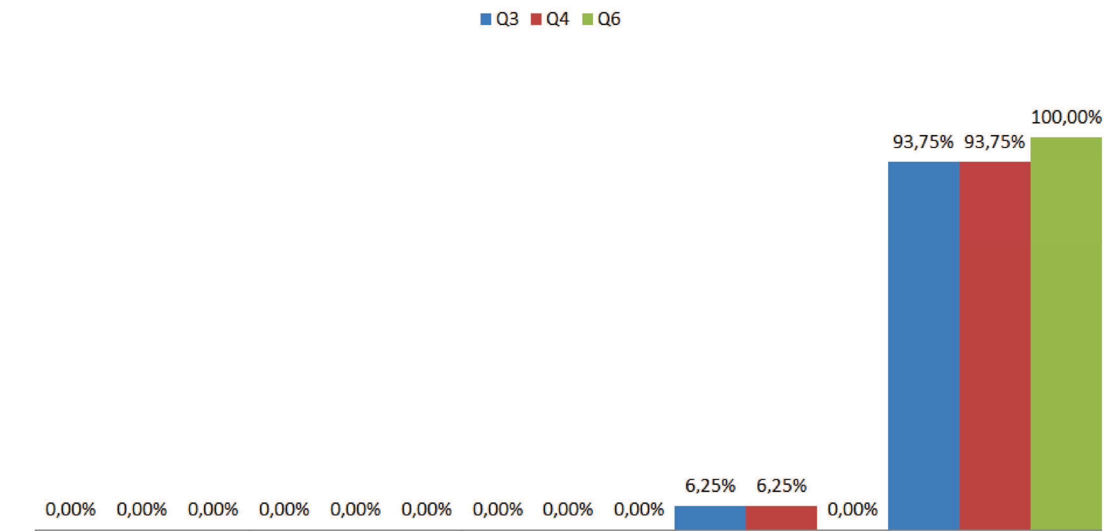
	Strongly disagree	disagree	neutral	agree	Strongly agrees
Q1	0,00%	0,00%	25,00%	12,50%	62,50%
Q2	0,00%	12,50%	31,25%	25,00%	31,25%

Figura 4.20: Q1,Q2

could it be useful for increasing security in returning to the periscope quota?;

- Question 6 - Could XR periscope be used to perform secondary tasks?.

As can be seen in Fig. 4.21, for Question 4, 93.75% of the users strongly agree and 6.25% agree that the XR Periscope would be helpful to increase the security in the procedure to return to periscope depth. This result agrees with Question 3, not rejecting Hypothesis 1. In the same table, in Question 6, 100% of the users strongly agree that the XR Periscope can be used to perform secondary tasks. Secondary tasks are the ones described in Hypothesis 2, not rejecting it either.

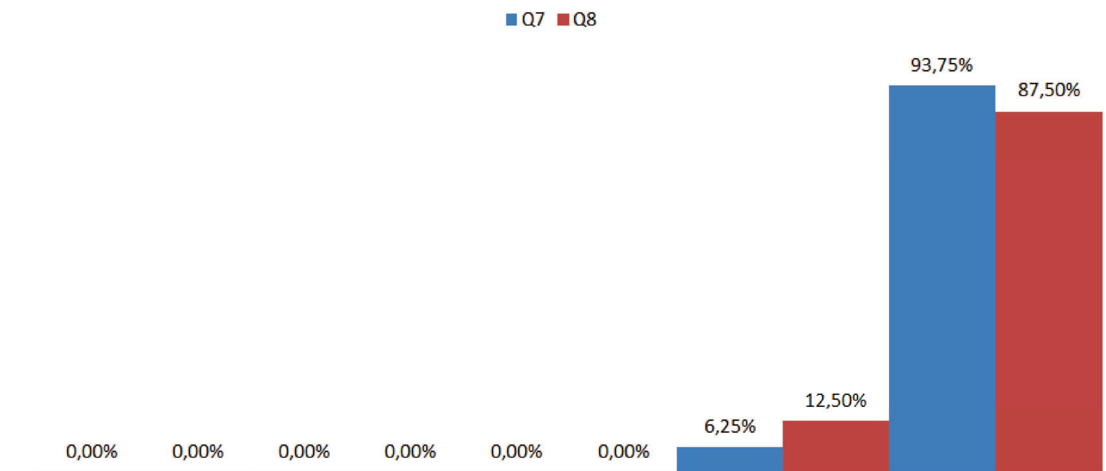


	Strongly disagree	disagree	neutral	agree	Strongly agrees
Q3	0,00%	0,00%	0,00%	6,25%	93,75%
Q4	0,00%	0,00%	0,00%	6,25%	93,75%
Q6	0,00%	0,00%	0,00%	0,00%	100,00%

Figura 4.21: Q3,Q4,Q6

Question 8 asks: "Did the classification and distance of the contacts provided by the XR Periscope help compile the tactical scenarios ?". It is a direct answer to Hypothesis 3, and as can be seen in Fig. 4.22, 87.5% of the users strongly agree and 12.5% of the users agree, not rejecting this hypothesis.

In Question 7, 93.75% of the users strongly agree and 6.25% agree that the XR Periscope increases navigation security.



	Strongly disagree	disagree	neutral	agree	Strongly agrees
Q7	0,00%	0,00%	0,00%	6,25%	93,75%
Q8	0,00%	0,00%	0,00%	12,50%	87,50%

Figura 4.22: Q7,Q8

Question 9 asks if the information is clearly displayed. Since 87.5% of the users strongly agree and 12.5% agree with this question, we assume that the interface is user-friendly. At last, Question 5 asks if the use of the XR Periscope helped in the decision-making process in the exercise, and 75.0% of the users strongly agree and 6.25% agrees with that question, showing that the equipment can be useful in the decision-making process. The results for Questions 5 and 9 can be seen in Fig 4.23.

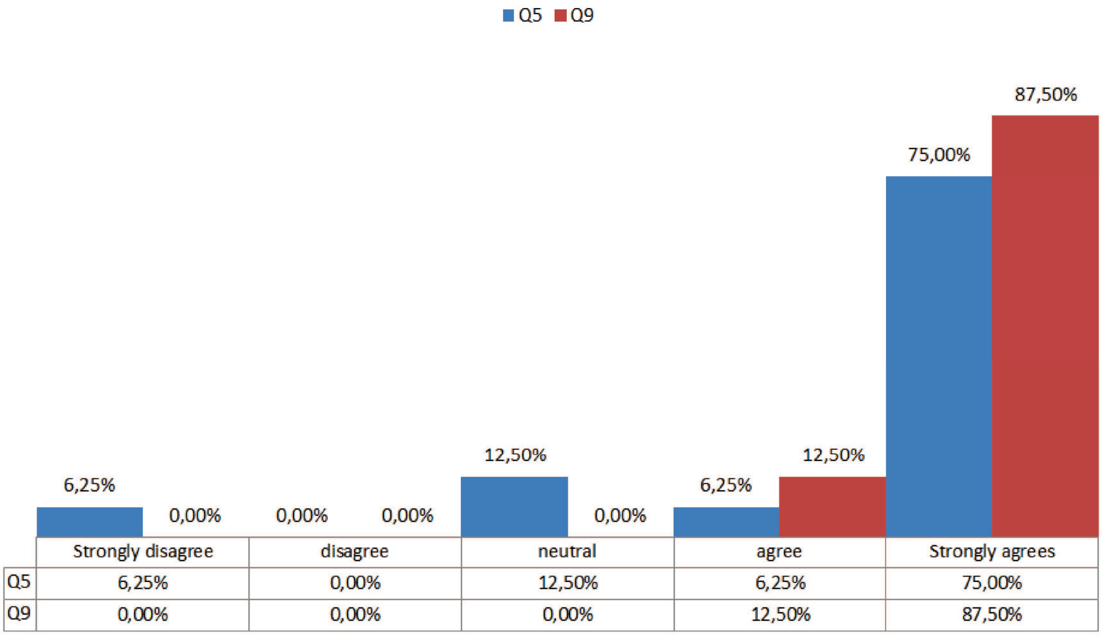


Figura 4.23: Q5,Q9

Capítulo 5

Conclusions and Future Work

5.1 Conclusion

In this work we proposed a novel periscope solution, based on XR and computer vision technologies. Our user experience tests confirmed that our approach can be an efficient and in the neat future substitute the traditional submarines periscopes.

In relation to the computer vision features, our results were able to confirm that artificial datasets were a viable alternative to the intent of performing object detection and classification in this specific scenario, where there are no real datasets available yet. Additionally, this type of dataset showed to be less costly and faster to produce, easier to manipulate and label than real images. Since the presence of simulators in the Navy force is consolidates, we claim that our solution allows more precise and robust datasets in the future.

The Data Aquisition Module (DAQ) functions give us great possibilities to improve the trained detection and classification model as we will be able to train the CNN using the artificial images together with real data. The DAQ automatizes the tasks of collecting and labeling data, collecting real images at different real-life conditions that otherwise would be very difficult to manage.

Although we present the complete solution for a real submarine operation, our work still lacks tests in a real submarine at the sea. Those test would be really costly due to the harsh environment and high pressures that the submarine operate, an operation procedure of the equipment must be studied and authorized by the submarine force yet.

Even though the existence of possible bias on the user experience tests, like a low FPS rate, not being able to test in the real environment and the fact that the developers were present during the test, The obtained results confirm that the XR periscope solution can be very useful to improve the safety of a submarine conduction, considerably increasing its operational efficiency by reducing the submarine discretion fee. It can also bring a profound revision on the "perisher" technique by using disruptive technology as it helps the commander decision process.

Our technique can be adapted and used in other vessels and cases, like port entrances, monitoring points of interest at sea, or as part of the control system of automatic vessels. The rendering of other vessel classes can be easily implemented and included in the dataset of synthetic images to increase the spectrum of the detection and classification model.

Besides that, other functionalities can be easily developed, such as the estimation of the closest approach point, bow angle, and GODEX [3], which is the maximum period of time that the submarine can stay at periscope depth without risk of collision with other vessels, which is the maximum period of time that the submarine can stay at periscope depth without risk of collision with other vessels. Those functionalities depend on a series of parameters such as ship speed and direction, other vessels' directions and draught, adaptation to different conditions like night vision, infrared images, etc.

Referências

[1] Database, howpublished = `https://drive.google.com/drive/folders/1mi2bxucd6n9zmuqbnzcgyv8zv9v_nlnn?usp=sharing`, note = Accessed: 13 September 2021.

[2] AutoIt Consulting Ltd. AutoIt scripting language. Available online, 2021. Accessed on June 2nd, 2021.

[3] Bastos, R. Development of a web-based periscope simulator for submarine officer. *Naval Postgraduate School Monterey CA: Monterey, CA, USA, 10* (2014).

[4] Bisong E. Google colaboratory. in: Building machine learning and deep learning models on google cloud platform. apress, berkeley, ca., 2019.

[5] Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y. M. YOLOv4: optimal speed and accuracy of object detection. Available online, 2020.

[6] BRASIL, Comando da Força de Submarinos. ComForS-730: procedimentos operativos para submarinos, 2012. in Portuguese.

[7] CIRM. AmazÔnia azul. Available online, 2021. Accessed on June 2nd, 2021.

[8] CIRM. Comissão interministerial para os recursos do mar (cirm). Available online, 2021. Accessed on June 2nd, 2021.

[9] Crewman on a US Navy ASW helicopter. from Honolulu, Hawaii, Public domain, via Wikimedia Commons. Uss key west at periscope depth, 2004. [Online; accessed August 19, 2021].

[10] De Mello, J. P. V.; Tabelini, L.; Berriel, R. F.; Paixão, T. M.; de Souza, A. F.; Badue, C.; Sebe, N.; Oliveira-Santos, T. Deep traffic light detection by overlaying synthetic context on arbitrary natural images. *Computers & Graphics 94* (2021), 76–86.

[11] Estado-Maior da Armada. EMA-305: doutrina militar naval, 2017. in Portuguese.

[12] Grabowski, M. Research on wearable, immersive augmented reality (WIAR) adoption in maritime navigation. *Journal of Navigation 68*, 3 (2015), 453–464.

[13] International Maritime Organization. Listing of current IMO publications. Available online, 2021. Accessed on June 2nd, 2021.

[14] Juliani, A.; Berges, V.-P.; Teng, E.; Cohen, A.; Harper, J.; Elion, C.; Goy, C.; Gao, Y.; Henry, H.; Mattar, M.; Lange, D. Unity: A general platform for intelligent agents, 2020.

[15] Kanjir, U.; Greidanus, H.; Oštir, K. Vessel detection and classification from spaceborne optical images: A literature survey. *Remote Sensing of Environment 207* (2018), 1–26.

[16] Kim, K.; Hong, S.; Choi, B.; Kim, E. Probabilistic ship detection and classification using deep learning. *Applied Sciences 8*, 6 (2018), 936.

[17] KOLLMORGEN. Technical manual for the kollmorgen model 330 periscope, 2005.

[18] Laera, F.; Fiorentino, M.; Evangelista, A.; Boccaccio, A.; Manghisi, V.; Gabbard, J.; Gattullo, M.; Uva, A.; Foglia, M. Augmented reality for maritime navigation data visualisation: a systematic review, issues and perspectives. *Journal of Navigation 74* (05 2021), 1–18.

[19] LeCun, Y.; Boser, B.; Denker, J.; Henderson, D.; Howard, R.; Hubbard, W.; Jackel, L. Backpropagation applied to handwritten zip code recognition. *Neural Computation 1* (1989), 541–551.

[20] Lee, S.-J.; Roh, M.-I.; Oh, M.-j. Image-based ship detection using deep learning. *Ocean Systems Engineering 10* (12 2020), 415–434.

[21] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. L. Microsoft COCO: common objects in context. In *European Conference on Computer Vision (ECCV)* (2014), pp. 740–755.

[22] Liu, Y.; Zhang, Y.; Zuo, S.; Fu, W.-T. Boatar: a multi-user augmented-reality platform for boat. pp. 1–2.

[23] Milgram, P.; Takemura, H.; Utsumi, A.; Kishino, F. Augmented reality: A class of displays on the reality-virtuality continuum. *Telemanipulator and Telepresence Technologies 2351* (01 1994).

[24] O Periscopio Magazine. Perifoto de uma fragata tipo 23componente da força naval inimiga, 2020. [Online; accessed June 06, 2021].

[25] Open Source Vision Fundation. OpenCV: open source computer vision library. Available online, 2021. Accessed on June 2nd, 2021.

[26] Pan, S.; Yang, Q. A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on 22* (11 2010), 1345 – 1359.

[27] Pan, S. J.; Yang, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering 22*, 10 (2010), 1345–1359.

[28] Redmon, J. Darknet: open source neural networks in C, 2016.

[29] Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: unified, real-time object detection. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV, USA, 2016), IEEE, pp. 779–788.

[30] Redmon, J.; Farhadi, A. YOLO9000: better, faster, stronger. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI, USA, 2017), IEEE, pp. 6517–6525.

[31] Redmon, J.; Farhadi, A. YOLOv3: an incremental improvement, 2018.

[32] Rios, C. R. "ações de submarinos", 2019.

[33] Shao, Z.; Wu, W.; Wang, Z.; Du, W.; Li, C. Seaships: A large-scale precisely-annotated dataset for ship detection. *IEEE Transactions on Multimedia 20* (08 2018), 1–1.

[34] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)* (San Diego, CA, USA, 2015), , p. 14.

[35] Stanton, N. A.; Roberts, A. P. J.; Fay, D. T. Up periscope: understanding submarine command and control teamwork during a simulated return to periscope depth. *Cognition, Technology & Work 19*, 2 (2017), 399–417.

[36] Ward, C. M.; Harguess, J.; Hilton, C. Ship classification from overhead imagery using synthetic data and domain adaptation. In *OCEANS 2018 MTS/IEEE Charleston* (2018), pp. 1–5.

[37] Wikipedia, the free encyclopedia. Likert scale, 2014. [Online; accessed July 07, 2021].

[38] Wikipedia, the free encyclopedia. Sensorama, 2021. [Online; accessed August 31, 2021].

[39] Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks?, 2014.

APÊNDICE A – User Experience Questionnaire

The following questionnaire, fig. A.1, was applied on 19 submarine officers on the 09 and 10 of August 2021.

the questions are in Portuguese.

PERGUNTAS DE AVALIAÇÃO:
Responda numa escala de 1 a 10 de acordo com suas impressões.

1. Grau de experiência com o procedimento apresentado (VH).

NADA MUITO

2. Grau de experiência com realidade virtual.

NADA MUITO

3. O XRP auxiliaria na compilação do quadro tático.?

NADA MUITO

4. Assumindo que o XRP possa ser lançado a partir da cota de segurança, ele poderia ser útil para o aumento da segurança no retorno a cota periscópica?

NADA MUITO

5. No quadro tático simulado, o XRP influenciou a decisão de manobra?

NADA MUITO

6. O XRP poderia ser utilizado para realização de tarefas secundárias?

NADA MUITO

7. O XRP aumentaria a segurança da navegação?

NADA MUITO

8. A classificação e distância dos contatos, fornecida pelo XRP, auxiliou na compilação do quadro tático?

NADA MUITO

9. As informações apresentadas estão de fácil entendimento?

NADA MUITO

10. O senhor visualiza alguma outra utilidade para o XRP?

Comentários

NOME:____

Data: _____/AGO/2021

Figura A.1: User Experience Questionnaire - in Portuguese