DM

Dashboard for Collecting and Depicting the Marine Megafauna Presence

MASTER DISSERTATION

Paulo Filipe Nunes Aguiar
MASTER IN INFORMATICS ENGINEERING

UNIVERSIDADE da MADEIRA
A Nossa Universidade
www.uma.pt

December | 2020

Dashboard for Collecting and Depicting the Marine Megafauna Presence

MASTER DISSERTATION

Paulo Filipe Nunes Aguiar
MASTER IN INFORMATICS ENGINEERING

ORIENTATION
Marko Radeta

CO-ORIENTATION
Filipe Magno Gouveia Quintal

Abstract

While more and more technologies and software are being created and applied for the ocean setting, most of them still remain at high cost, and hinder the data to wider public. Understanding the marine biodiversity can be achieved through numerous ways, however, there is a lack of consensus and operability when depicting the marine megafauna population. Moreover, Deep Learning (DL) techniques are becoming accessible to wider population, and there is a potential of exposing them to the marine biologists, involving them to participate in public web-based dashboards, depicting those data.

This dissertation addresses such issues, by providing an interactive dashboard, capable of facilitating the classification, prediction and deeper analysis of marine species. Using the State of Art (SoA) Machine Learning (ML) techniques for image vision, and providing the interactive visualizations, this thesis seeks to provide a less cumbersome apparatus for marine biologists, who can participate further in data gathering, labelling, depicting, ecological modelling, and potential calls for action.

In further, this dissertation document provides the aquatic dashboard functionality using Human-Computer Interaction (HCI) techniques and interactive means to ease the upload, classification, and visualization of collected marine taxa, with a case study on marine megafauna imagery (e.g. whales, dolphins, sea birds, seals and turtles). As it will be hereinafter described, marine biologists, as end users, will evaluate of the proposed dashboard.

Keywords: Biodiversity Assessments · Marine Megafauna · Deep Learning · Back-end/Front-end Development · Information Visualization.

Resumo

Todos os dias surgem novas tecnologias e softwares que podem ser aplicados no ecossistema marinho, sendo que a maioria destas permanecem com um custo elevado, dificultando assim o acesso ao público em geral. O conhecimento deste sistema e de toda a biodiversidade nele existente, pode ser alcançado de diversas formas, no entanto, existe uma falta de consenso e operacionalidade ao descrever a população de megafauna. Além disto, técnicas de aprendizagem automática como o *deep learning*, permanecem acessíveis a uma população mais ampla, e existe o potential do envolvimento de profissionais da área, mais conhecidos como biólogos marinhos, para participar na criação e usabilidade de plataformas conhecidas como *dashboards*.

Esta tese tem como função debater estas questões, fornecendo um *dashboard* interativo, capaz de facilitar a classificação, previsão e análise mais profunda das espécies marinhas. Usando técnicas de aprendizagem automática de última geração, para informação visual em imagens, e fornecer interfaces visuais muito interativas, esta tese procura fornecer uma ferramenta simples para os biólogos marinhos, podendo assim participar na recolha de dados, rotulagem, modelação ecológica e possíveis pedidos de alerta.

A dissertação produzirá um *dashboard* funcional, utilizando técnicas de Interação Humano Computador (HCI) e meios interativos para facilitar o carregamento de dados, a classificação e visualização da fauna marinha coletada (p.ex. baleia, golfinho, ave marinha, foca e tartaruga). Como será descrito durante este manuscrito, biólogos marinhos, como utilizadores finais, irão participar na avaliação deste proposto *dashboard*.

Palavras-chaves: Avaliações da Biodiversidade · Megafauna Marinha · Aprendizagem Automática · Desenvolvimento Web · Visualização de informação.

Acknowledgements

And after six years, my academic journey comes to an end. After all this years, I have met a lot of important people that I must thank for being part of this big journey and always incentive me when I needed the most.

First I would like to thank my parents and my sister. Thank you for always supporting me, trusting and rooting for me, thank you for all the unconditional love.

Because life would be nothing without friends, I would like to thanks all my friends for always guiding and supporting me and most importantly turning this journey a lot more fun!

Last but not least, I would like to thank all the professors that were part of this journey but most importantly my supervisors, Prof. Marko Radeta (Wave Lab Coordinator) and Prof. Filipe Quintal for all the dedication, advice, patient, commitment, and sharing knowledge that made this dissertation possible.

This study is part of LARGESCALE project with grant no. PTDC/CCI-CIF/32474/2017 by Fundação para a Ciência e a Tecnologia (FCT) and Portuguese National Funds (PIDDAC). It is also supported by the funding by project UID/EEA/50009/2019 and UIDB/50009/2020 (ARDITI/LARSyS) as well as by Programa de Cooperación INTERREG V-A España-Portugal MAC (Madeira-Azores-Canarias) 2014-2020, throughout project INTERTAGUA (Interfaces Aquáticas Interativas para Deteção e Visualização da Megafauna Marinha Atlântica e Embarcacoes na Macaronésia usando Marcadores Rádio-transmissores), with Ref: MAC2/1.1.a /385. Moreover, conducted studies were supported by the additional funding from project MITIExcell - EXCELENCIA INTERNACIONAL DE IDT&I NAS TIC (Project Number M1420-01-01450FEDER0000002), and project SUBMARINE (Simulators for Understanding the Biodiversity using Mobile and Augmented Reality as Interactive Nautical Explorations), provided by the Regional Government of Madeira. Also, it was supported by the project INTERWHALE - Advancing Interactive Technology for Responsible Whale-Watching, with grant no. PTDC/CCI-COM/0450/2020 by Fundação para a Ciência e a Tecnologia (FCT). Finally, project is supported by the Wave Lab[1] - interactive technologies for depicting the anthropogenic impact on marine biosphere.

Paulo Filipe Nunes Aguiar

[1] http://wave.arditi.pt

Table of Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
CNN	Convolutional Neural Networks
*CO*2	Carbon Dioxide
CSV	Comma-separated Values
DL	Deep Learning
DOM	Document Object Model
DTL	Data Transformation Language
EO	Earth Observation
EoL	Encyclopedia of Life
ESA	European Space Agency
GDP	Gross Domestic Products
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HARP	High-frequency Acoustic Recording Package
HCI	Human-Computer Interaction
HTTP	HyperText Transfer Protocol
IoT	Internet of Things
JSON	JavaScript Object Notation
mAP	Mean Average Precision
MHD	Mobile Hand-held Device
ML	Machine Learning
MSE	Mean Square Error
MVC	Model-View-Controller
NASA	National Aeronautics and Space Administration
NRT	Near Real-Time
OOM	Oceanic Observatory of Madeira
OpenCV	Open Source Computer Vision Library
REST	REpresentational State Transfer
RPC	Remote Procedure Call
SA	Software architecture

$SatEO$ Satellite Earth Observation

SoA State of Art

SQL Structure Query Language

SUS System Usability Scale

$SWIR$ Short-Wave Infrared

$t-SNE$ t-Distributed Stochastic Neighbor Embedding

TF TensorFlow

UAV Unmanned Aerial Vehicle

UI User Interface

URL Uniform Resource Locator

USD United State Dollar

VHR Very High Resolution

1 Introduction

Nowadays, there is a considerable request in understanding and studying the marine biodiversity. This ecosystem provides a lot of resources for human lives and has a estimated market value of marine and coastal resources of $3 trillion USD per year, about 5% of global GDP. Marine biological diversity, also known as marine biodiversity can be defined as the abundance and richness of species in the world's oceans and seas [32]. Such ecosystem represents three quarters of Earth's surface (75%) and represents 99% of the living space on the planet by volume, containing nearly 200,000 identified species, however the actual number may reach millions [75].

Marine biodiversity is therefore an important field, because it is known that more than 3 billion humans are dependent on the Earth's resources for their livelihoods. However, the human footprint has a significant cost in this ecosystem. Oceans are being heavily hit by a lot of pollution from depleted fisheries, loss of coastal habitats, carbon dioxide (30% absorption of the carbon dioxide produced by humans) and other human activities. All of these actions represent as much as 40% of the oceans being polluted [75]. These results in ecosystems are becoming more resistant to providing the benefits for the planet [8].

To address these challenges, this dissertation will spotlight the marine megafauna, and typical species which occur as local and migratory population in Atlantic region (e.g. whales, dolphins, turtles, seals and sea birds). It will further, support in assessing their population using the collected imagery, providing an interface to the collected data from diverse sources (e.g. sea vessels, IoT devices, citizen scientists, aerial imagery, etc).

As most of the collected data will be images, this dissertation will enhance the SoA and explore the diverse ways of identifying the presence of marine megafauna using deep learning (DL) [22], capable to perform multiple neural networks for image classification and object detection. Proposed apparatus will in further, provide an interactive interface using Information Visualization [2], facilitating the search queries and overall the user experience.

1.1 Motivation

Understanding biodiversity can be performed through numerous activities. From whale watching to bird watching, where numerous people attend these events. While marine megafauna watching promotes important educational, environmental, scientific, and other socioeconomic benefits, it is also a $3.95 billion USD industry [72], attracting more than 13 million people a year, in 119

[2] Information visualization is the process of representing data in a visual and meaningful way so that a user can better understand it [83]

countries worldwide [42]. Such activity is largely used for tourism marketing of coastal and marine protected areas, communities, regions and countries. Nevertheless, this activity is prone to several challenges, including the necessary fuel costs and CO_2 emissions, as it depends on the exact areas where most of these species are located. Still, traditional techniques for detecting cetaceans are being performed using the observational surveys from the coast, mostly using the binoculars [38], which remains the sole and expensive technique.

There are other passive remote sensing ways to collect the data about biodiversity. For instance, an initiative by ESA, where the SPACEWHALE service is proposed. SPACEWHALE is capable of semi-automatically detecting the large whales from Very High Resolution (VHR) satellite imagery. Such service is capable of benefiting from distributed non-governmental organizations, academic and private companies who are engaged in global studies of monitoring efforts on large whales. The final report of this service are collected data of specific types of whales detected from the space, formatted data in a CSV file [20]. However, all of such collected data remain hindered to the general public, and focus solely on the one type of a whale specie, which can be easily discriminated from space.

Another big concern, as the Related Work section will present, is the absence of tools, capable of depicting and interactively exhibit a way to understand the ecosystem as a whole. This is perhaps due to the reason that most of the gathered data is handled by large companies, hidden from the general public, while being of significant price cost. Nevertheless, there are systems which involve public and the open source data gathering and depicting, as in the work by the Monicet platform [55]. This example is an attempt in exploring the imagery of collected whales and dolphins in Azores[3] islands. However, this system's core features relies exclusively on whales, and more specifically on flukes of sperm whales and other greater whales. Therefore, taking into account all the aforementioned issues, this dissertation seeks into providing a more complete system, addressing wider imagery, marine taxa and wider data input (from people, IoT devices, satellite imagery, UAV's, etc).

1.2 Research Questions, Objectives and Contributions

The main contribution of this dissertation is therefore to collect, interpret and depict the imagery of marine megafauna in a web- based dashboard. There are several challenges which will be addressed to achieve such contributions:

[3] Azores also known as Açores is an archipelago composed of nine volcanic islands in the North Atlantic Ocean. About 1,360 km (850 mi) west of continental Portugal [78]. Azores is also known for being a hotspot of marine biodiversity.

(i) **[RQ1]. How a software can be used to decipher and depict the marine megafauna presence?**

The challenge in this research question is to advance the State of Art (SoA) in current software techniques used for collecting and classifying the marine megafauna, performing the near real-time (NRT) AI pipeline.

(ii) **[RQ2]. How well does the image vision performs when estimating marine megafauna?**

It will also provide an interface for object detection, allowing the marine biologists to confirm the confidence rates of the proposed software predictions (e.g. using galleries, grid views and data visualization in back office[4]);

(iii) **[RQ3]. Which is the feedback from marine biologists and other stakeholders when using this system?**

Through user studies, this research will measure the impact of depicted marine megafauna datasets and imagery, and understand to which extent do the marine biologists actually find the software to be supportive or not.

Objectives of this dissertation are therefore to:

(i) **[O1. COLLECT]** Design and develop a robust back-end, capable of receiving the data and images through API routines from diverse input: mobile phones, IoT devices and Satellite imagery.

(ii) **[O2. INTERPRET]** Produce a Machine Learning (ML) pipeline to classify and predict the species from the obtained data, providing the accuracy of the prediction inside of the proposed platform.

(iii) **[O3. DEPICT]** Portray the collected predictions in front-end, where marine biologists can participate in human-in-the-loop detection and verification, and finally re-training of the marine megafauna models.

Contribution of this dissertation is therefore, in the novel interactive platform capable to retrieve, classify and portray the collected imagery from: **(i)** Sea vessels (side imagery collected by the participants and tourists on-board whale watching trips); **(ii)** Drones or IoT (aerial surveys); and **(iii)** VHR images (satellite) inputs. Therefore, with the gathered proposed apparatus, this thesis will evaluate and classify the marine megafauna, returning the result of the Artificial Intelligence (AI) algorithm, adding the images to the gallery, and allowing them to be assessed at the later stage. The proposed system is a web-based dashboard module, facilitating means to depict and alert the presence of marine megafauna in NRT. It will focus on treating, gathering and presenting

[4] The back office is the portion of a company made up of administration and support personnel who are not client-facing. Using functions like information technology, human resource management, finance and accounting [44].

the data in visually appealing way, increasing the user experience ,and providing the answers to marine biologists' queries.

1.3 Structure of the Document

This dissertation outline is the following:

(i) **INTRODUCTION.** This chapter provided the motivation and research questions which will be addressed when designing a module to assess the occurrence of marine megafauna. High-level perspective of problems and potential solutions are presented, including the opportunities to explore means to collect, store, classify and depict the marine megafauna data.

(ii) **RELATED WORK.** The second chapter describes the SoA and related work in image vision algorithms, including the means to assess the marine biodiversity, as well as the current interactive dashboards and ways of depicting the collected imagery, present in both market and scientific communities.

(iii) **METHODOLOGY.** The third chapter describes the proposed apparatus, consisting from: **(i)** Back-end development software; **(ii)** Image vision classification (using object detection algorithms) pipeline and used models for recognizing whales, dolphins, birds, seals and turtles; and **(iii)** Front-end interactive component, serving to explore the collected data. Moreover, study setup is described in order to verify the obtained mdoel accuracy, usability and overall HCI performance of the dashboard when using such system.

(iv) **RESULTS.** The fourth chapter yields the results of usability and HCI tests, as well as the overall obtained feedback from the marine biologists. It will also provide findings of used classifiers, their accuracy and agreements with the marine biologists' labeling, when discriminating the marine megafauna.

(v) **DISCUSSION.** The fifth chapter discusses the findings and obtained results from feedback and training models, outlining constraints and providing future works for software application in depicting and alerting the marine megafauna presence in NRT.

(vi) **CONCLUSION.** The final chapter provides the overall thesis findings, outlining the research contributions, answering to the proposed research questions.

2 Related Work

Since obtaining the footage from the sea-vessels remains at the high cost, this dissertation will study the potential of crowd sourcing to collect the imagery. Moreover, as there is a lack of public dashboards capable to collect, interpret and depict the collected imagery of marine megafauna, this section covers the current state of the art techniques to assess the imagery of marine megafauna from remote distance. Moreover, prior work by scholars using such technologies is described, as well as an overview of the image vision algorithms which support the classification (object detection) of marine megafauna.

2.1 State of the Art

To discriminate the differences and select the most appropriate affordable setup, this section presents the techniques capable of assessing the imagery data. Moreover, cost required to gather this imagery will be the most important aspect in each technique, therefore supporting our approach of using crowd sourcing, having persons collecting the imagery by themselves. For this reason, the dissertation describes three scenarios to assess the imagery (land, sea and space, as in Section 2.1.1). Furthermore, an overview of the best practices in image vision is described, suggesting the best practices and pipelines for classifying the taxa (Section 2.1.2). Finally, an overview of the interactive visualization techniques, serving to facilitate the access to these imagery to wider public is presented (Section 2.1.3).

2.1.1 Collect: Techniques for Assessing Biodiversity

At the moment, there are three possible ways to assess the marine megafauna biodiversity, using land, sea and space collected imagery. Below, an outline of technologies and techniques are presented.

2.1.1.1 Land: On-shore Crowd sourcing

As aforementioned, land can be used for mapping the megafauna presence. Using visual surveys and land observation methods have been a regular activity of citizens throughout generations [43]. The crowd that cooperates in spotting the presence of biodiversity mostly used powerful binoculars[5] from the coast. When these entities spot the potential existence of marine megafauna, they dispatch

[5] Typical Gear Specifications (e.g.): Prisma type, Power 18X, Objective Lens 50mm diameter, Filter Size 50mm diameter, Real Field of View 3.7 degrees, Field of View in 1000 meters 65 meters, Weight 1180 grams [9]

this knowledge to other entities, skippers on board the sea vessels [38]. See figure 1 as an example of typical land-based whale-watcher.

Fig. 1. Example of a On-shore Whale Watcher [38]

This particular labor is indeed crucial to support locating the megafauna as well as to assist the logistics of sea vessels reaching the whale watching spots. Guiding sea-vessels and mostly catamarans interested in whale-watching is an essential effort also to guarantee the safety of the animals, as well as to allow the exploration of the nearby landscape areas [16].

In fact, technologies for remote monitoring of species are becoming more affordable and are emerging. This types of technologies seem to provide promising alternatives, because current methods require expensive and complex setup. Nevertheless, Hills *et al.* [39], introduced a low-cost IoT apparatus with adaptive methods to gather data. The author named this solution as AudioMoth[6], providing an answer to the expensive and complex gears to listen to cicadas[7], where the end user is part of the community, and can deploy such tool and capture ultrasonic frequencies. As a study, the author deployed 87 AudioMoths at four locations New Forest National Park[8] for 2- to 3-moth periods. As a result of this study, the author achieved a true positive rate of 98% and a false positive rate of 1% when discriminating Cicadas. With these results in mind, this dissertation agrees

[6] AudioMoth is a low-cost, full-spectrum acoustic logger, based on the Gecko processor range from Silicon Labs. Just like its namesake the moth, AudioMoth can listen at audible frequencies, well into ultrasonic frequencies. It is capable of recording uncompressed audio to microSD card at rates from 8,000 to 384,000 samples per second. [6]

[7] The cicadas are a superfamily, the Cicadoidea, of insects in the order Hemiptera (true bugs).Cicadas have prominent eyes set wide apart, short antennae, and membranous front wings. They have an exceptionally loud song, produced in most species by the rapid buckling and unbuckling of drumlike tymbals. [80]

[8] New Forest National Park is located on the south-central coast of England and lies within the county of Hampshire. Covering an area of about 220 square miles, the New Forest National Park takes in approximately 15 miles of coast between Calshot Castle (most easterly point) and Hurst Castle (most southerly point), and stretches northwards to Whiteparish and westwards to Ringwood. [73]

that AudioMoth provides additional opportunities for users with limited budgets to identify and analyze species.

Therefore, given the aforementioned possibilities how megafauna can be assessed from the land, it seems appropriate to have a pipeline to advise and cooperate with other entities interested in whale-watching, as more data improve the general public experience. This effort from the citizens is decisive to support this thesis application, and thus it should assist the citizens with the more fine grain resolution of data, depicting the data publicly in the web-based dashboard.

2.1.1.2 Sea: Marine Biologists Field Trips

For many decades, sea has been an important point to explore the marine biodiversity. However, using citizens to collect data for scientists, and combining these data with machine learning (DL), can be more beneficial to obtain more precise data about the population of one specie. For instance, Cetalingua [11] asked the general public to collaborate, however the data is not accessible for non membership. On the contrary, Happywhale [36] project, sends volunteers to boat rides to collect data. They ask citizen to take photos and reports of mammal encounters, providing a way to assess and identify species offline.

The obtained Catalogue from Oceanic Observatory of Madeira (OOM[9]) [60] provides an important effort in classifying the individual species by given dorsal fins of the whales and dolphins. This catalogue is of particular interest as it offers the rich dataset of cetacean dorsal fins and flukes, observed from a boat in Madeira Island. Authors use photo-identification, and manually annotate the dorsal fins by drawing the trailing edges around them, which is laborous in terms of time. They assess diverse cetaceans spotted in 14 years catalog, in order to help assessing the population of whales and in further, bridging the gap between marine biology findings and the general public [4]. An example of the catalog image is depicted in figure 2.

[9]OOM is a science unit within Autonomous Region of Madeira that focus their resources in studying the population ecology of cetaceans in Madeira, given the biological, cultural and socio-economic importance of this taxonomic group for the region. [60]

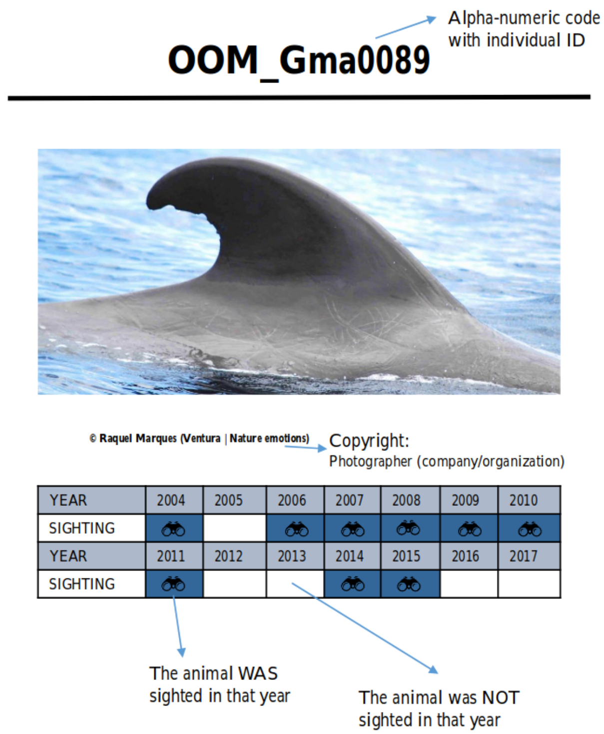


Fig. 2. Example of the Catalog item with dorsal fin of pilot-whale off-shore Madeira coast [4]

In overall, identification and marking of the animals can be performed in several ways: [25]

1 **- Machine Learning (ML) Algorithm Detects the Specie**

In this step, collected data are:

(i) Stored in datasets to train computer vision models;

(ii) Model is used to detect individuals whales and dolphins in photos;

(iii) Confidence rate is shown for the species.

2 **- Algorithms to Identify Individuals**

After each taxa is reported with the latitude and longitude, algorithms make digital "footprints" of each animal, where occurrence patterns can be observed. This is possible with trailing edge annotation and detection, where a fluke or a dorsal fin of the specie is being colored, manually, to increase the contrast of the image, necessary for the algorithms. This potential of AI, replaces hours of human labor for minutes of computer visualization, scanning for matches across tens of thousands of photos. For instance, figure 3 provides an example how a collected fluke of whale and odontoceti[10] can be used later on for analysis to identify a specie with the help of machine learning (ML) algorithms, respectively.

[10] Odontocetis are toothed whale. This species are a parvorder of cetaceans that includes dolphins, porpoises, and all other whales possessing teeth, such as the beaked whales and sperm whales. [74]

3 **- Population Dynamics Define Conservation Action**

Conversely, marine biologists observe the collected dataset and understand the movement ecology. Using statistical analysis, population, model size and migration are used to generate new insights and support data-driven conservation action (e.g. biopsy sample collection, sighting reports, conservation action, etc).

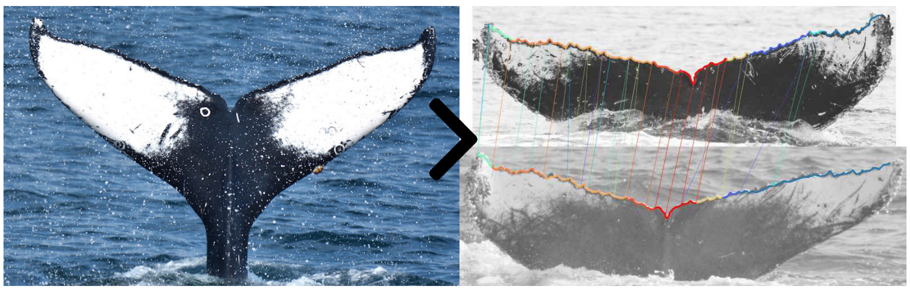


Fig. 3. Example of the gathered data and running Artificial Intelligence [36]

While the aforementioned techniques are useful means to identify the whales against the other details in an image, and produce an important catalogue of diverse species, however, these procedures still remain expensive [26]. This imposes more marine biologists to go on a boat ride, and approach the species to perform manual identification analysis. Moreover, collected data using the tourists (and not marine biologists and citizen scientists) might be noisy and cumbersome to interpret for future training of models [28].

2.1.1.3 Space: Earth Observation Imagery

As previously mentioned, SPACEWHALE [21] offers a bridge between space and earth, Earth Observation (EO). This service consist in a machine-learning algorithm that used CNNs, which are widely used in modern applications such as facial or vocal recognition [67]. For every algorithm of this type, it is necessary to train it with big datasets. In the example of ESA, the company behind this service trained with a enormous dataset of VHR digital aerial imagery from aircraft-based surveys. An aerial image with 2cm resolution is used and then down-sampled to 31cm to match WorldView 3 [11] satellite imagery. In figure 4, an example of the process of acquisition of satellite imagery is depicted.

[11]WorldView-3 enhances the industry-leading constellation with the addition of 8 high-resolution SWIR bands as well as the 12 bands of CAVIS [18]

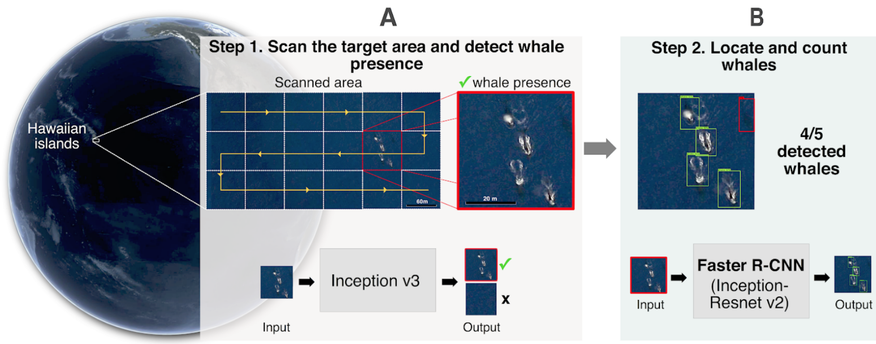


Fig. 4. Guirado Approach - CNN Whales from Space [19]

From the aforementioned image, it is understandable that with more objects being identified, the more robust the algorithm is. At the final of this process, a report (.CSV formatted data sheets and SQL databases) is generated, providing an overall abundance and distribution of the marine species that have been examined with the help of ESAs expert team of reviewers.

Summarizing this technique, it focuses on collecting analytical data with the help of machine learning (ML) algorithm, by providing the constant feed of satellite imagery. Their program provides a significant utility to understand more about the ocean and marine life. However, their reported service remains solely applied for their personal usage and the service does not provide any downstream application layer. In this dissertation, a dashboard application is provided which allows the experts marine biologists to train the models of marine species, while larger audience may upload and visualize the prediction and accuracy of the uploaded imagery.

2.1.2 Interpret: Image Vision Best Practices

Thanks to the recent advances in technology, namely computer hardware and software, algorithms have been developed to support processing images [62]. Such algorithms emerge and evolve over the years with the introduction of Artificial Neural Networks (ANNs). ANNs can be defined as a structure of connected artificial neurons, also known as nodes. This computational model is capable of performing massively parallel computation for data processing, therefore aiding in modeling complex real-world problems [7]. However, because image vision techniques are challenging to understand thoroughly and the current available source code is cumbersome to be obtained, newer algorithms are created on a daily basis. Due to the difficulty to adapt and shape existing algorithms for different purposes [76], several popular techniques were developed:

(i) **Machine Learning (ML)**

Nowadays, AI is a popular topic. This area of science is a an important branch of computer science, however IT solutions using this approach are crucial for every section of activity, due to the increase in performance of productivity and consequently, the company profit. AI has a high demand in every sector, from algorithms to analysis (data science) [58]. Machine Learning (ML) is an AI application which continuously learns and improves automatically from previous experience without being explicitly programmed. Such process of learning starts with observation of data such as examples or direct experience. As a result of repetitive experiences, patterns in data and decision making are spotted. The primary aim of this technique is to allow computers to learn by themselves, without the aid of human resources [22].

(ii) **OpenCV**

Computer vision is popular among data science students and professionals. This topic focus in transforming data with a specific goal, from a video or camera into either a decision or new representation (e.g: face recognition/ID in modern smartphones). Using the aforementioned technique, a library called OpenCV was released in January 1999. This framework plays an important role in the evolution of computer vision. OpenCV helps users to develop projects by providing a computer vision with ML infrastructure. This technology aim is to support developers and researchers to build vision applications as fast as possible, therefore providing a simple-to-use computer vision infrastructure, tailored to their needs [51,84].

(iii) **Deep Learning (DL)**

Simultaneously, deep learning is a sequence of the machine learning. When the need for a more accurate and better algorithm appeared, a new type of ML was defined, known as Deep Learning. This new ML type allows computational models with multiple layers to learn representations of data with multiple levels of abstraction. As a result of this technique, the ANNs drastically improved (e.g. speech recognition, visual object recognition, object detection). Deep learning discovers intricate structures in large datasets by using the back propagation algorithm[12]. This technique indicates how a machine should change internal parameters that are being used in the computation calculus of each layer. In other words, the machine enhances the parameters used to calculate the previous layer, improving the next iteration layer. Therefore, learning from the previous layers (back propagation algorithm) [45]. DL has become central to the ML efforts of every present multinational company (e.g. Facebook, Microsoft and Yahoo).

(iv) **TensorFlow**

As aforementioned, ML has driven advances in numerous different sectors, hence providing a need for better techniques. However, algorithms related to this topic are tough to be built from scratch. As a consequence, in 2016 TensorFlow was released by Google Brain Team [1]. Multiple updates were done and TF 2.0 was released. This framework is not a technique but a enabler to ease the deployment and implementation of all techniques aforementioned.

[12]provides a popular method for the design of a multi-layer neural network to include complex coefficients and complex signals [47]

Conversely, as this system was a success, Google launched a open-source project with a goal to challenge the previous existing systems. This provides a precious tool for everyone and accepting external community contributions, which resulted in the production of a more robust apparatus. Comparing TensorFlow with the previous techniques/technologies this framework is based on a branch of AI called Deep Learning. Additionally, this framework allows an individual to build its own AI application with low amount of time. There are more machine learning frameworks like TensorFlow (e.g. fchollet, Keras, Lasagne, Blocks, Pylearn2, Theano. see table 1 for a more detailed comparison). Moreover, this framework has the ability to run multiple instances of its solution (using multiple machines). Also, Google uses a distributed system version, while releasing an open-source version that was only available on single node machines. However, a new version was released to the public with a multi-GPU feature. With this simple configuration, TensorFlow gave the best outcome for complex problems, such as stock trade predictions, NASA star classification, traffic management, and other [56].

Table 1. Popular Machine Learning Frameworks @ Google Trends

	Producer	**Contributers**	**Type ((Un)Supervised)**	**Multicore (CPU/GPU)**	**Popularity (%)**
TensorFlow	Google	Open	Both	Both	83
Apache Spark	Apache Software Production	Open	Unsupervised	Both	82
OpenCV	Intel	Open	Both	GPU	63
Pytorch	Based on Torch	Open	Unsupervised	Both	41
Keras	Tsung-Yi Lin	Open	Supervised	Both	53
Theano	Montreal Institute for Learning Algorithms	Open	Supervised	Both	1

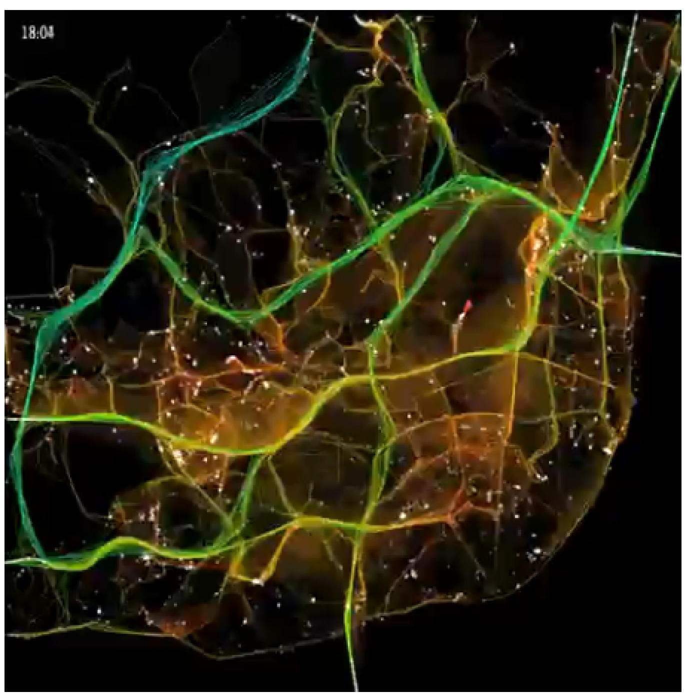
Given the diverse ML techniques, this thesis will opt for using TensorFlow in software architecture (SA) for the following reasons: **(i)** Flexibility (being highly modular approach, therefore rapid in pipeline integration); **(ii)** Portability (no need for extra hardware, thus being a low cost solution); **(iii)** Research and Production (can be used to train and serve models in live mode with customers, in this case, marine biologists); and **(iv)** Performance (allows to make the best of the used hardware with support for threads, asynchronous computation and queues) [56].

2.1.3 Depict: Interactive Information Visualization

Given the advances in IT, complex and multidimensional data are being generated at an uncommon rate. While data is crucial to the knowledge and advances of multiple research areas, this imposes a challenge on data analysts [3]. This section provides the outline of the current techniques capable of digesting and simplifying the overloading of data in a interactive and user-friendly manner, therefore improving the user experience when exploring the corpulent datasets.

Nowadays, every IT individual understands the influence of JavaScript[13] in web solutions. This popular language has grown and created more than 100 frameworks [13]. Therefore, it is not a big surprise that one of interactive solutions uses such library. Mainly, D3.js[14] is a JavaScript library that allows application of data-driven transforming in the document through the Document Object Model (DOM) [54]. This framework provides smooth transitions and vast interactions while creating and fetching big data. D3.js is capable of increasing the user experience significantly, while portraying the required data in novel ways, by virtue of the already defined visual representations, counting with more than 100 interactive examples. Another similar approach, also using the popular programming language JavaScript, is Highcharts [37]. Highcharts just like D3.js deliver out-of-the-box source-coded interactive solutions, however with somewhat less documentation.

Using the aforementioned concept, Pedro Cruz[15], developed diverse ways to display data without overloading users with information. The author focused in simple and minimalist representation, while building novel concept designs. In the example below, figure 5, it is possible to see the author's depiction of complex information (e.g. Lisbon's Traffic).



[13]JavaScript is a scripting or programming language that allows you to implement complex things on web pages [24]

[14]D3.js is a JavaScript library for manipulating documents based on data [54]

[15]Pedro Cruz is a data visualization specialist and explorer. Recently he was awarded for Design and Social Movements by the 4th Ibero-American Biennial of Design. He won the ACM SIGGRAPH Student Research Competition in 2010 and was a semi-finalist in 2011 [10]

Fig. 5. Lisbon's Traffic by Pedro Cruz [10]

As it was aforementioned in figure 5, the author did not showcase a top-view, using aerial or satellite imagery, to represent the traffic. Instead, the author focused on a simple and visually appealing user-interface, portraying with density and color codding the affluence of traffic. In another study by McCandless [53], the author showcases a diversity of techniques and concepts for future implementations. Author focused on portraying inspiration for future projects. Comparable to the aforementioned study, Design Density Labs [17] also provide diversified illustrations for future integration.

As it was displayed in this section, there are several techniques, illustrations and concepts in providing interactive information visualization, however, not much work yields the visualization and simplification of marine imagery and machine learning applied to marine megafauna. All things considered, this dissertation will use such methods and concepts to portray big data, using D3.js. This choice was reflected based on the popularity and documentation available as well as on the nonexistence of current platforms portraying marine biodiversity using interactive techniques.

2.2 Prior Research

In this section, existing techniques and technologies will be explored and presented. Firstly, in section 2.2.1 platforms collecting marine megafauna data. Second, in section 2.2.2, existing software for marine megafauna. Third, in section 2.2.3, existing technologies portraying the marine biodiversity. Lastly, in section 2.2.4 a brief summary of this dissertation direction and choices will be presented and explained.

2.2.1 Collect: Assessing Marine Megafauna

As aforementioned, technologies capable of collecting, identifying and estimating the size of the marine megafauna already exist. One of them used Satellite Earth Observation (SatEO) approach [33,34], where authors pointed out the usefulness of CNNs to outperform humans in visual tasks such as image classification and object detection. This research studied cost-effectiveness and the accuracy of identifying one group of cetaceans (Minke whales[16]). In their study, authors used Google Earth images raising the prediction accuracy by 36%, compared to the baseline detection models. Authors stressed the importance for a need of more high-resolution pictures.

Regarding the collection of marine data, there are several other works and platforms portraying marina megafauna, see table 2. As aforementioned, OOM keeps track of the marine megafauna

[16] The minke whale, or lesser rorqual, is a species complex of baleen whale. The two species of minke whale are the common minke whale and the Antarctic minke whale [81]

presence in their catalogue, focusing on data collection of flukes and dorsal fins. Since this is valuable to the overall understanding of the marine biodiversity abundance, they also create a website portraying all gathered pictures from the sea vessels [61]. In essence, this platform is a rich data inquiry, where the data can be used in future tools, such as the one provided in this dissertation. In the aforementioned catalog by OOM, additional work should be addressed in creating a easier and appealing user interfaces, aiding marine biologists in classifying the species.

Table 2. Platforms Portraying Cetaceans Presence and Identification

	Techniques	**Data Type**	**Accuracy**	**Cost**	**Access**
Guirado	CNN	Space	0.94	Expensive	Closed
HappyWhale	Image Vision	Vision	N/A	Cheap	Open
Cetalingua	DL	Land/Sea	N/A	Medium	Open
Monicet	Image Vision	Vision	N/A	Medium	Open
OOM	Manual	Land/Sea	N/A	Medium	Open

2.2.2 Interpret: Image Vision in Biodiversity

Estimating marine biodiversity is also essential for the conservation of wildlife [82]. However, conventional approaches remain expensive and sometimes do not reach a wider population. Therefore, automatic count through VHR satellite imagery could lead to advances and provide new opportunities. Conversely, as it is mentioned by Guirado *et al.* [34], CNNs have a enormous effectiveness in image vision solutions. Another apparatus was also proposed by Haimeh [35] as finFindR[17]. This open-source software displays an image recognition solution to interpret individual species in simple steps without much effort. The author proposed a simple and intuitive desktop application, where the users select an image and click a button for trailing edge of the dorsal fin. With such elementary process, the application returns a specie estimation, based on similar dorsal fin match, existing in the database. However, the proposed solution only covers a small portion of the species and focus solely on dorsal fins of the species. Therefore, this thesis will build upon by scaling to diverse marine megafuna, and whole-body species estimation.

In a similar approach to finFindR, DARWIN[18] was proposed by Stanley [68] in 1995. This is another desktop application focusing in image vision through already existing identified dolphin fins. In contrast to finFindR, this solution requires more effort to: **(i)** Load the image into the software **(ii)** Trace outline of the fin **(iii)** Locate special features of the fin **(iv)** Query the already existing database for identification [15], while returning identical results. As for the user interface,

[17] finFindR is an R and c++ based library for doing photo recognition on dolphins, sharks and other [35]

[18] DARWIN is a computer vision system that helps researchers to identify individual bottlenose dolphins[19], Tursiops truncatus, facilitating the comparison of digital images of the dorsal fins of new dolphins with a database of previously identified dolphin fins [15]

this system looks more old fashioned and slightly cumbersome to handle. In brief, DARWIN also contributes to cataloging marine megafauna assessments, however, open-source solutions may be more important in reaching the general public.

Moreover, similarly to the aforementioned works, a few following this concept grew popular, each of them with their respective pros and cons, being portrayed in this section. FinBase [2] advantages relies on a **(i)** Automate digital image management activities; **(ii)** Portability and maintenance; and **(iii)** Attribute-based approach (Dorsal fin notches or marking). Another solution is FinScan [65] this one entrust on **(i)** No batch tracing function; **(ii)** Requirement of user to manually traces fin; and is **(iii)** Quick to validate (less than 1 minute). Similar to findFindR, CurvRank [5] focus on **(i)** Fin contour extraction (edge as ordered coordinates); and **(ii)** Trailing edge extraction and integration in Flukebook. OBIS SEAMAP [14] is another apparatus that focus on **(i)** Fin matching; and **(ii)** Detection of movement (tracking a specie route). Different from the previous systems, Photo-ID Ninja [29] capability focus on transforming any imagery from cetacean or group, cropping to a fin based picture. Therefore the system capabilities consist of **(i)** Automatically crop dorsal fins; and **(ii)** Intensive manual processing (about 1 hour to validate a individual). Distinctive from the previous tools, Dolphin Matching Using Google Cloud Vision (DMUGCV) [50] was develop in collaboration with google's AI engineers. Google AutoML enable this framework to identify individual dolphin in seconds versus the average 1-20 hours (depending in fin distinctiveness). Therefore when using in a enormous collection (10,000 images each year), the minimum annually time save is 4550 hours. All things considered, the system pros consist of **(i)** a cloud-based; **(ii)** automated matching system based on computer vision and machine learning; **(iii)** Using trailing edge; where there are **(iv)** No limit to catalog size; in a **(v)** Stand-alone matching program (not part of an integrated database system). However, this apparatus are still in a beta-testing, therefore not freely available to the general public. All things considered, a better understanding of this solutions can be comprehended in the following table 3.

Table 3. Technology Softwares to produce Image Vision in Marine Biodiversity

	System Type	**Matching Focus**	**Average Matching Time per Photo**
CurvRank	Automated	Dorsal Fin	<1 minute
Photo-ID Ninja	Semi-Automated	Dorsal Fin	∼1 hour
DMUGCV	Automated	Dorsal Fin	<1 minute
Finscan	Semi-Automated	Dorsal Fin	N/A
finFindR	Semi-Automated	Dorsal Fin	N/A
finBase	Semi-Automated	Dorsal Fin	∼1 hour
DARWIN	Semi-Automated	Dorsal Fin	N/A

2.2.3 Depict: Understanding Biodiversity Interactions

Setting aside from interpreting the images, when dealing with depicting the data, dashboards are an extremely popular way to represent data and can be found mostly online. Few *et al.* [23], define dashboard as a visual display of the the most important information needed to achieve one or more objectives that have been consolidated on a single computer screen, so it can be monitored and understood at a glance. In simple words, dashboards should provide all required information on a simple screen, without overloading the user with information. For this reason, the failure of a implementation is related to the design, which in return can provide insufficient or overloaded information. In this section, a variety of existing dashboards and techniques for depicting the biodiversity will be discussed.

(i) **Involv (Application using Voronoi Treemaps)**

Horn *et al.* [41], presented an apparatus using an information visualization technique, the Voronoi Treemap[20] algorithm. The author apparatus is known as Involv, a multi-touch interactive visualization of the Encyclopedia of Life (EoL), marking available all information about life present on Earth. See figure 6 for an example.

Fig. 6. Involv system - Using Voronoi Treemap Algorithm [41]

As a result, Involv seems to perform well in certain areas of the taxonomy. However, when the number of edge crossings are small, the visualization starts to become obtrusive and burdensome to interpret. Moreover, in other taxonomy areas, this approach is less successful, because

[20]Introduced in 2005, the Voronoi treemap algorithm is a information visualization technique that visualizes hierarchical data by recursively partitioning convex polygons using weighted Voronoi diagrams [59]

of the higher number of nodes and levels. Due to the reason of dense clusters of siblings species, overlaying each other, therefore perturbing the interactive visualization. Nevertheless, these technique, provides Involv a satisfying user interface, where the user's queries are represented in a smooth and interactive way.

(ii) **Monicet**

Established in 2009, Monicet [55], is a platform where companies related to cetacean observation in Azores collect and share taken photographs of the cetacean flukes, see figure 7. This platform is managed by a research group of biodiversity in Azores. Monicet platform has the share data and crowdsourcing philosophy, where users can support them with the collected imagery, allowing the benefits for both whale watching companies and tourists.

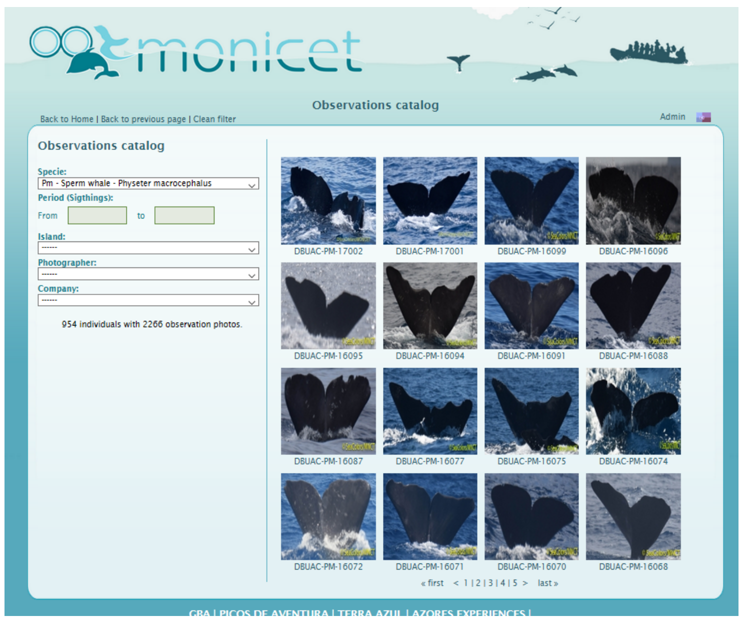


Fig. 7. Monicet Catalogue [55]

(iii) **Pattern Radio: Whale Songs**

While understanding the marine biodiversity can be achieved through numerous ways, Chen *et al.* [12] approach focuses in using acoustics to assess whale sounds and depicting it in a web interface. For the web interface, the author limited to show the collected data with a play button. As for the methodology used to gather the data, the author used custom-build recorders called high-frequency acoustic recording packages (HARP)s and hydrophones[21], that were affixed to the ocean floor. These devices start recording cetaceans calls and human activities (defined as noisy data). However, since these devices have a limited capacity, when the batteries run out or the memory is full, they go to the location and retrieve the devices to obtain back the collected data. While this solution seems interesting, the technology used lacks accessibility,

[21] Underwater Microphones

due to the limitation of storage capacity and time to retrieve the data. As a result, the gathered data is already outdated. As for the depicted solution, the web interface presented by the author lacks the use of information visualization techniques to capture the generic public attention, providing more interactivity and appealing functional visualizations for explorations.

(iv) **Google - Vision AI**

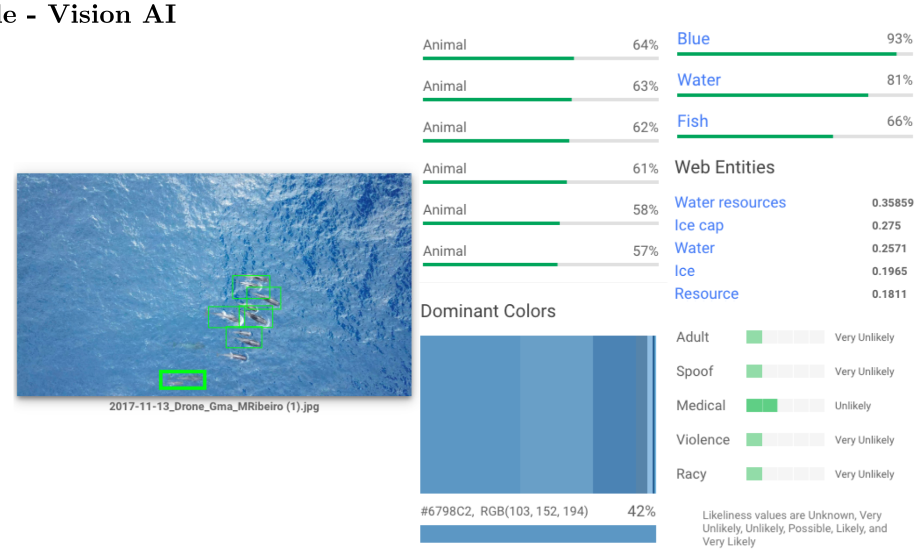


Fig. 8. Google's Vision AI - Misleading Results

A service developed by Google called, Vision AI [31,30], offers several options to integrate computer vision models into the applications and web sites. Moreover, this service is split in two computer vision products: **(i)** AutoML Vision; and **(ii)** Vision API. This dissertation will focus on Vision API, considering that the used approach will be similar to this one. Google's Vision API offers powerful pre-trained machine learning models through REST and RPC APIs[22]. Assigning labels to images and classifying, therefore detecting objects. As a result, when using web images which already exist online, they provide their URL location (if contained in image tags), however when using external images, their system struggles. See figure 8 as an example of the aforementioned results.

As it is shown in figure 8, Google's Vision AI has some flaws when depicting the data. The results from a private uploaded image, showing a group of dolphins, are identified as animals, from 57% to 64% of accuracy. With this in mind, this Dissertation acknowledges that the existing proposed solution is interesting and useful when using web images, however for a target audience of marine biologists, their approach lacks effectiveness.

(v) **Google AI Experiments: Bird Sounds**

Yet another work from the same company provided other ways of depicting the data. Acoustic diversity emerged as a possible indicator of biodiversity, due to numerous practical advan-

[22] API is a software intermediary that allows two applications to talk to each other without need to know a lot about each other. Just by using simple a request and reply [66]

tages. Using passive acoustics allows to explore habitats that are usually challenging to access, therefore the required capital for solutions that use this method is less than the others [27]. In similar approach, Tan *et al.* [70], focused on gathering bird sounds and depicting the data in a interactive way. The author collected bird sounds and used ML to organize them. To depict these data, author used a technique called (t-SNE[23]). In comparison to the aforementioned apparatus from Chen *et al.*, the solution illustrated by Tan *et al.* offers an interactive and visually appealing way of representing enormous chunks of data.

(vi) **INaturalist**

As it is common knowledge, Mobile Hand-held Devices (MHDs) are very popular nowadays [49]. MHDs have enormous processing power, thus far making these devices a powerful tool to daily tasks. Consequently, developers start targeting these devices with software (mobile applications). Mobile applications have grown over the years and are crucial to every individual daily life basics. From custom analyses tools[25], games and social networks to simple things like calculator, camera filters, calendar, etc. With a growing market [40] and the need of extending the understanding of marine biodiversity behaviour and life, a mobile application called INaturalist [52] was developed. This application creates a community, where its users interact with each other learning about close by species encounters. The application core concept is simple, the user can find nearby species to see while participating in a guide tour, for instance, when encountering a specie the user capture a picture, upload it and share within the community, see Figure 9. Therefore cataloguing new entry in a open-source global database. INaturalist have grown quickly and is one of the most popular applications featuring biodiversity, counting with more than 400.000 scientists and users. All things considered, the concept used by this apparatus is indeed important for this dissertation future choices.

[23]t-Distributed Stochastic Neighbor Embedding, also known as t-SNE is a technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional datasets. The technique can be implemented via Barnes-Hut[24] [77] approximations, allowing it to be applied on large real-world datasets [57]

[25]Google integrated tools (e.g Google Fit, Google Calendar, Google Maps, Google Drive, etc)

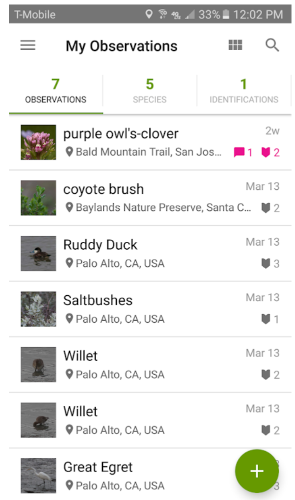


Fig. 9. Inaturalist - User Interface

2.2.4 Summary of the Related Work

Given the aforementioned techniques and existing resources, this dissertation will pursue the low-cost solution in providing and depicting the big datasets to wider stakeholders, interested in understanding the dynamics of marine megafauna population. Therefore, a balance between the diverse techniques will be available (e.g. side imagery - data from sea vessels; and aerial imagery - data from IoT devices). Moreover, a statistical record will be possible, allowing an overview of the collected data and visualizing such reports and alerts. The photo-identification will be performed by the ML analysis, using TensorFlow, object detection, allowing the images of the reports to be interpreted, stored and displayed effectively in the provided dashboard. Simultaneously, for the user interface, a dashboard will be developed using Laravel[26] together with React Redux[27], preserving the existing WAVE[28] software architecture (MVC[29]). Finally, related to the aforementioned, while dashboards remain challenging to build, information visualization techniques will be thus used to adapt and enhance the user experience. This dissertation contributes by providing a novel way of visualizing such reports, while promoting and researching marine megafauna population.

---

[26]Laravel is a web application framework with expressive, elegant syntax [71].

[27]Redux is a predictable state container for JavaScript apps [64].

[28]WAVE is a R&D lab that focus in applying IT solutions to build interactive ways of portraying local marine biodiversity. This platform was developed for marine biologists as well as any individuals who wish to contribute with data or know more about the local marine megafauna. See `http://wave.arditi.pt/`

[29]MVC is an useful design pattern for designing interactive software systems [46]

3 Methodology

In this section, the dissertation document describes the required procedures and implementation steps to develop the proposed dashboard. First, in section 3.1, the creation of a AI model capable of identifying the marine megafauna (dolphins, whales, turtles, birds and seals) is depicted. Then, in the following section 3.2, the proposed dashboard and application of the aforementioned AI model is presented, describing its software, logic, and system architecture. Finally, in section 3.3, the study setup is described, presenting the location, obtained data inquiry, sample size, and the conducted experiments.

3.1 Machine learning

In this chapter, an overview of a neural network is described when training the models of marine megafauna species with deep learning. In first, section 3.1.1 describes the data collections and the used frameworks. Secondly, section 3.1.2 describes how the data annotation was performed on the existing image dataset. Furthermore, section 3.1.3, describes the used procedure to increase the existing dataset using sets of image manipulations. Lastly, section 3.1.4 depicts the process of training the models using the marine megafauna dataset. Following diagram depicted in figure 10 provides the understanding of steps for generating the AI Model.

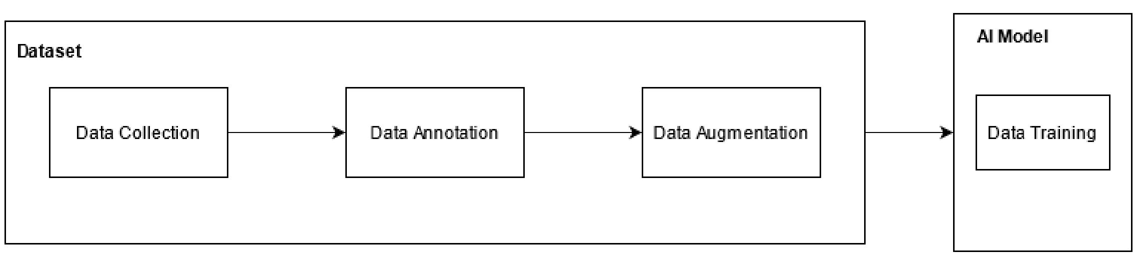


Fig. 10. Machine Learning procedure diagram

3.1.1 Data Collection

As aforementioned, this section provides an insight on how data was collected. For the production of the dataset, Supervise.ly [30] was used to gather, annotate, and augment the collected imagery. Image dataset was collected from the following sources: **(i)** google web search; **(ii)** google open images dataset; **(iii)** open images dataset V5; and **(iv)** private obtained imagery. Subsequently, all the gathered images were split into different folders, per specie, and type of image (side and aerial

[30]Supervise.ly is a web platform to manage, annotate, validate, and experiment with datasets & neural networks for computer vision [69]

view). Obtained imagery was sorted as (i) Mysticeti cetaceans, (ii) Odontoceti cetaceans, (iii) Sea birds, (iv) Pinnipeds, and (v) Turtles, which are frequent during the typical whale-watching trips. An overview of the number of obtained imagery is depicted in tables 4 and 5.

Table 4. Defined classes and respective image quantities (before the data augmentation)

	Number of Images	Number of Objects
Sea-Turtle	1787	2552
Sea-Bird	1660	4185
Odontoceti	2217	4487
Mysticeti	1966	2132
Pinniped	2380	6501
Total	**10010**	**13356**

Table 5. Dataset image sources

Source	Number of Images	Percentage of Images (%)
Kaggle Right Whale Recognition	1201	12.00
Google Image Search	5628	56.22
OID	3181	31.78
Total	**7630**	**100**

3.1.2 Data Annotation

Data annotation can be comprehended as the task of labeling data. In this dissertation, image recognition was the focus, and for a model to recognize a block of sensitive content as a dolphin, whale, turtle, sea-lion, or sea-bird it requires to label those classes in a large volume of annotated data to train the model. To annotate data, as aforementioned, we have used supervise.ly, this framework uses bounding boxes, see figure 11 on top of the sensitive content.

Fig. 11. Data Annotation of a Bird

3.1.3 Data Treatment

The next step after annotating all the data should be to train the model, however, several steps were taken to strengthen our model for the required uploaded imagery, overpassing the influence of weather conditions, refraction of water, noisy data, etc. Moreover, using transformation in the imagery allowed to further expand and clean the dataset, and prepare for the aforementioned conditions. Consequently, after pruning the dataset from duplicate imagery, five transformations using supervise.ly DTL were applied. Following algorithmms 1 and 2 provide the pseudo code for cleaning and augmentation, respectively.

Algorithm 1: Algorithm for data cleaning

```
consts: data, data_cleaned, W=800, H=600;
while data do
    if min_obj() => 1 then
        data_cleaned ← data.resize(W, H);
    end
end
return data_cleaned
```

Algorithm 2: Algorithm for data augmentation

```
consts: data, transf_data, data_aug, probability=0;
data_aug ← data;
data_transf ← data;
while data_transf do
    data_aug ← data_transf.flip_horizontal();
    data_aug ← data_transf.flip_vertical();
    if probability() => probability then
        data_aug ← data_transf.hue();
        data_aug ← data_transf.blur();
        data_aug ← data_transf.noise();
    end
end
return data_aug
```

Understanding of each transformation can be comprehended in the following figures:

(i) **Flip - Horizontally and Vertically**

With this particular transformation, the dataset was multiplied by three. Figures 12 and 13 showcase the transformation.

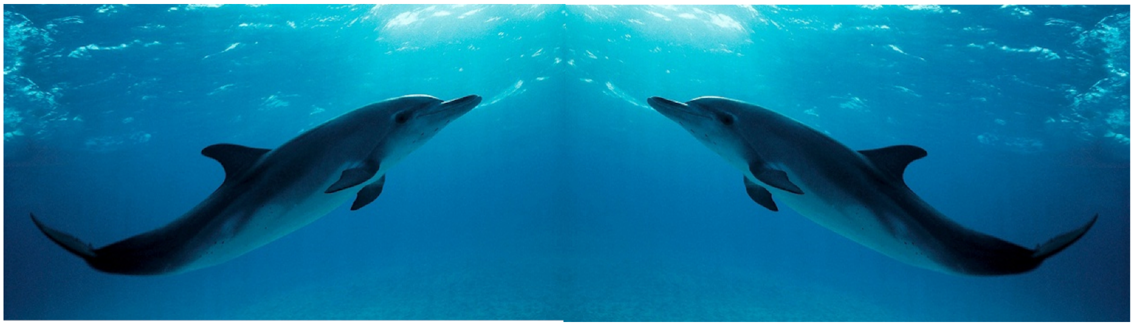


Fig. 12. Horizontal Flip Transformation

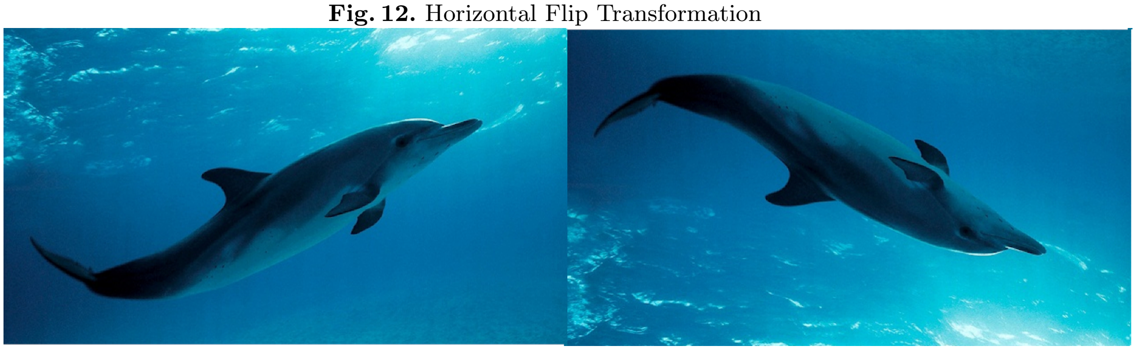


Fig. 13. Vertical Flip Transformation

(ii) **Hue - Color Saturation**

Color saturation also known as hue, as a transformation that intents to perform the same effect as sun light, causing a more or less vivid effect on images. Figure 14 depicts such transformation.

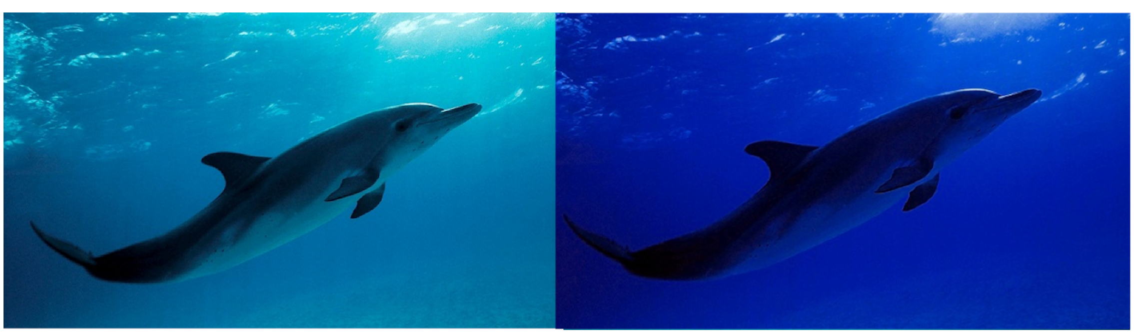


Fig. 14. Hue Transformation

(iii) **Blur**

This transformation is well known and avoided when processing videos or images for promotions, however the main rationale was to reduce the underwater effects. See figure 15 to understand the transformation.

Fig. 15. Blur Transformation

(iv) **Noise**

Lastly, the final transformation implemented was creation of noise. As aforementioned in blur, the same logic applies in this step. Figure 16 depicts this transformation.

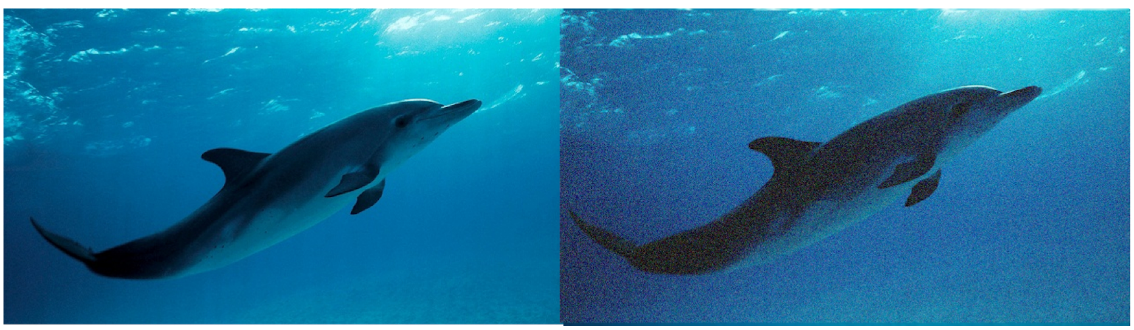


Fig. 16. Noise Transformation

Following these procedures, the dataset is downloaded and converted with the following algorithm 3, allowing the removal of unnecessary folders, removal of redundant data, with further organization of folders (train 99% of images, test 1% of images and labelmap).

Algorithm 3: Algorithm to clean and structure the dataset

---

consts: data, labelmap, train_img, test_img, annotations;

while *data* **do**

  $annotations \leftarrow data.annotation();$

  $labelmap \leftarrow data.getClasses();$

end

$nr\_imgs \leftarrow parseInt(data.size() * 0.99)$

$train\_img \leftarrow data(nr\_img);$

$test\_img \leftarrow data(-(data.size() - nr\_img));$

return $labelmap, train\_img, test\_img$

3.1.4 Data Training

As aforementioned, this chapter details the process of training the models using supervise.ly's composed datasets. To train the models, an open-source cloud service was used, Google Colab[31], due to the requirements for a more advanced power system specifications[32], being capable in providing faster output, hence more time for different model iterations and tests. This cloud service uses jupyter notebook[33], allowing the running blocks of code and feature markdown[34] language. Thus, this extensive process was structured into imperceptible steps (separation of concerns):

(i) **Verification of GPU Utilization**
Considering that this cloud service is open source, it comes with several constraints. For instance, multiple users are running their code on the same machines, and for this reason and to assure that it is possible to obtain a dedicated machine for ourselves, the first block of code is a simple print verification, whether or not the resources are available. With this output, if the GPU memory used is greater than zero, we disconnect and test to get another machine;

(ii) **Clone Git Repository and Setup the Settings**
Generally, IT solutions should be re-used and adapted. In this code block, an existing Git repository[35] is cloned, containing folders schematic and running scripts. Additionally, training variables were created, with the number of training steps and selected model;

(iii) **Installation of Required Packages**
In next, considering that scripts are running at unknown machine instances, it is necessary to install all the required packages, as well as to download the obtained dataset, moving them into correct folders. Eventually, after everything is set, the cloned object detection script starts running and the training begins.

(iv) **Exporting the Model and Running the Inference Test**
Finally, the training procedure is completed, where the output files are exported and converted into a single model file, using a TensorFlow converter, allowing the files to be converted into multiples formats (e.g. edgeTPU, tflite, tensorflowJS, etc). Generated model are then used in an inference test with different images from the training procedure.

[31] Google Colab is a free cloud service to develop deep learning applications using popular libraries such as Keras, TensorFlow, PyTorch, and OpenCV.

[32] Xeon Processors @2.3Ghz; 13GB RAM; NVIDIA Tesla K80; 33GB ROM(Free Space for user)

[33] Jupyter Notebook is a web-based interactive development environment for code, and data. Jupyter is flexible: configure and arrange the user interface to support a wide range of workflows in data science, scientific computing, and machine learning [63]

[34] Markdown is a lightweight markup language with plain-text-formatting syntax.

[35] `https://github.com/Tony607/object_detection_demo`

3.2 Research Apparatus

In this chapter, the full process of system usage and benchmark of the created model is explained. Software components encompassing front-end, back-end and Tensorflow pipeline are depicted, including the fluxogram, system architecture, obtained database and the Graphical User Interface (GUI) presented in the dashboard. The study setup describes the trained data sample, including the used scales and the tasks given to the users.

3.2.1 Software: the Tracker System

Hereinafter, User Interface (UI) implementation and back-end process is thoroughly explained. To fully understand the implementation of this system, it is structured into three sections:

(i) **Front-end**

As aforementioned, front-end contains three sections: (i) Collect; (ii) Interpret; and (iii) Depict. Each section was firstly designed and accommodated in the existing dashboard, using React Components, however, some of the pages were unnecessary too empty for a single feature and were later refactored into a single page containing all three features. Usage of React components was helpful in this refactoring step since it allows the calling of the components in any web pages. In this main screen, all three components were arranged, as depicted in figure 17.

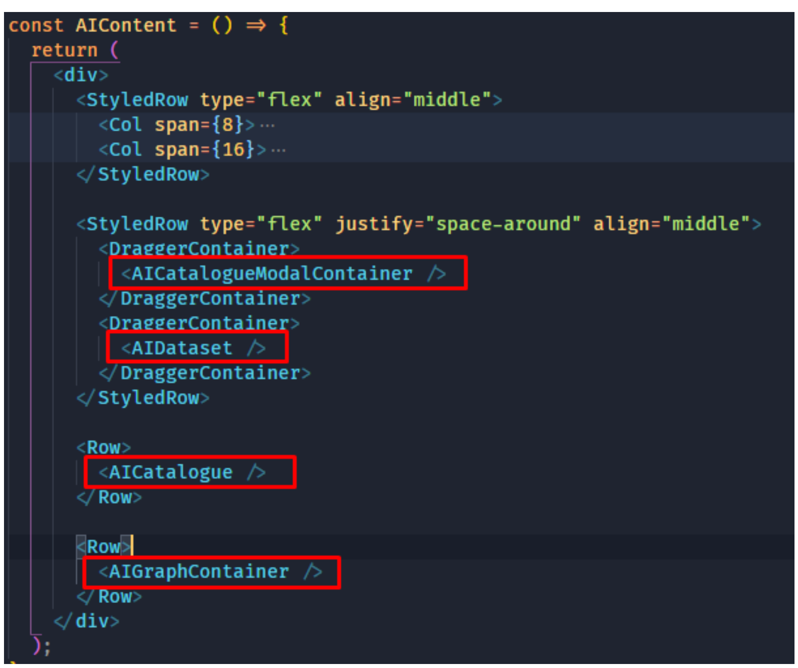


Fig. 17. React Main Component

The data to fill-in the page is obtained using state maps of react redux, being passed to the respective children components using react props[36]. An example is seen in the following figure 18. In following text, each component features are individually explained.

```
const mapDispatchToProps = (dispatch) => {
  return {
    fetchPhotos: (page, filters) => dispatch(fetchPhotos(page, filters)),
  };
};

const mapStateToProps = (state) => {
  return {
    data: state.trackerCatalogue.data,
    loading: state.trackerCatalogue.loading,
  };
};
```

Fig. 18. React map state to props

(1) **Collect**

Represented in the main component by *AIDataset*, this component was one of the main reasons for a single component to be created. The purpose of this component is to send images to a third-party website, supervise.ly. allowing the annotation to be performed ad-hoc. Images are uploaded to create a larger dataset, which will be used to re-train the model. For this reason, having a main component just with a drag-and-drop was not viable, therefore fitting all the developed components in a single React component was the obvious choice.

(2) **Interpret**

Represented in the main component by *AICatalogueModalContainer*, it is another upload function to the Wave server[37], serving to upload the imagery and send them to TensorFlow python script. The script returns the output in a pop-up modal, where user can agree whether or not to store the image and present it in the following gallery.

(3) **Depict**

Represented in the main component by *AICatalogue* and *AIGraphContainer*, this part contains the uploaded imagery in what is known as a carousel[38]. If selected, these images are further scaled fitting the user monitor resolution, allowing the easy-to-use navigated. A filter for individual species was created, returning the queried desired species, updating the previous interface. Statistics of the gathered data is also given, displaying the two graphs: (i) Average Accuracy (returns each species average accuracy of the uploaded imagery); and

[36] Props is a react keyword for properties where data can be manipulated between components easily
[37] `http://wave.arditi.pt`
[38] Display of images in a horizontal way.

(ii) Feature Time usage (returns the number of uploaded images per month in the last six months.)

(ii) **Back-end**

The back-end is responsible to return and complete all data for each request. In the proposed Tracker system, the back-end is no different. As aforementioned, the front-end receives and manipulates the data using react state maps, where all such data is attained from APIs. The following APIs endpoints were implemented:

(1) Return of all previously uploaded images, with respective TensorFlow output. (GET Method: api-detection)

(2) Return of specific species; (GET Method: api-detection/specie, being the specie a wildcard)

(3) Image detection and subsequently database insertion; (POST Method: api-detection);

```
Route::apiResource('ai-megafauna', 'API\TrackerController');
Route::apiResource('ai-detection', 'API\AiDetectionController');
Route::apiResource('ai-label', 'API\AiLabelController');
Route::apiResource('ai-model', 'API\AiModelController');
```

Fig. 19. Tracker API Endpoints

(4) Statistical data for both aforementioned graphs, GET Method, as seen in figure 20

```
Route::prefix('ai')->group(function () {
    Route::get('specie', 'API\AnalyticsController@aiSpecie');
    Route::get('history', 'API\AnalyticsController@aiHistory');
});
```

Fig. 20. Statistics API Endpoints

(iii) **Tensorflow Pipeline**

This implementation is crucial for all the previously described processes. When the user uploads an image, the image is further processed using this pipeline, being responsible to receive an image, run TensorFlow inference test python script for object detection, gather all the required information to fill-in the database, including: (i) Image name - path; (ii) Label - the name of the identified species; (iii) Accuracy - how often predictions equals the labels; and (iv) Coordinates of the bounding box, including xmin, ymin, xmax, and ymax. All of these are returned to the back-end, allowing any desired model to be compaired using JSON output values.

3.2.1.1 Fluxogram

A more detailed explanation of the proposed interpretation process is further explained. First, users are required to register in the existing *WAVE*[39] platform. Second, users perform login and

[39] `http://wave.arditi.pt`

access the dashboard. In the dashboard, side-panel is chosen with the Tracker AI section. On this page, the individual is invited to upload a desired image, where the image is sent to the described TensorFlow pipeline, returning all the information to the front-end, shown in a modal pop-up. Nevertheless, to block undesired content (noisy data), this step is preformed only when there is a high enough accuracy (e.g. more or equal than 70% match). If the accuracy reaches the threshold limited (0.7), the further dialog box pops-up, showcasing the results from the Tensorflow. User then chooses to agree or disagree with the provided accuracy. If former, the uploaded image will be uploaded to the server and displayed in the gallery. In latter case, the image gets removed from the interface and no further action is required. A fluxogram depicting the procedure can be comprehended in the following figure 21.

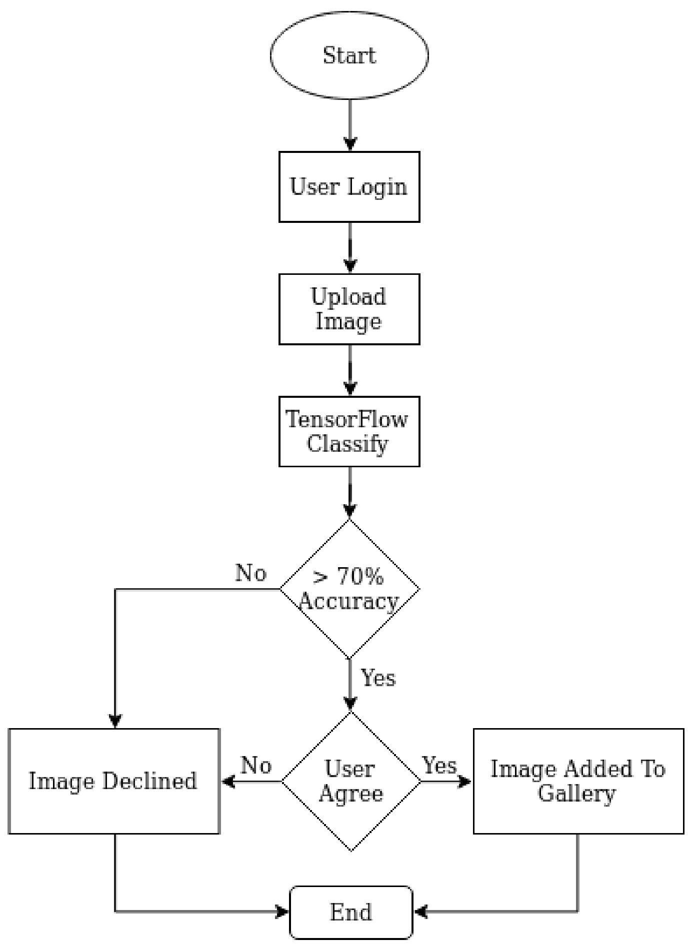


Fig. 21. Fluxogram explaining the image detection process

3.2.1.2 System Architecture

The overall system architecture is based on existing web server, allowing robust API calls to manipulate the data. These calls are secured by the architectural principle of open/close and the software design pattern MVC, assuring the further access security to the data, seen in figure 22.

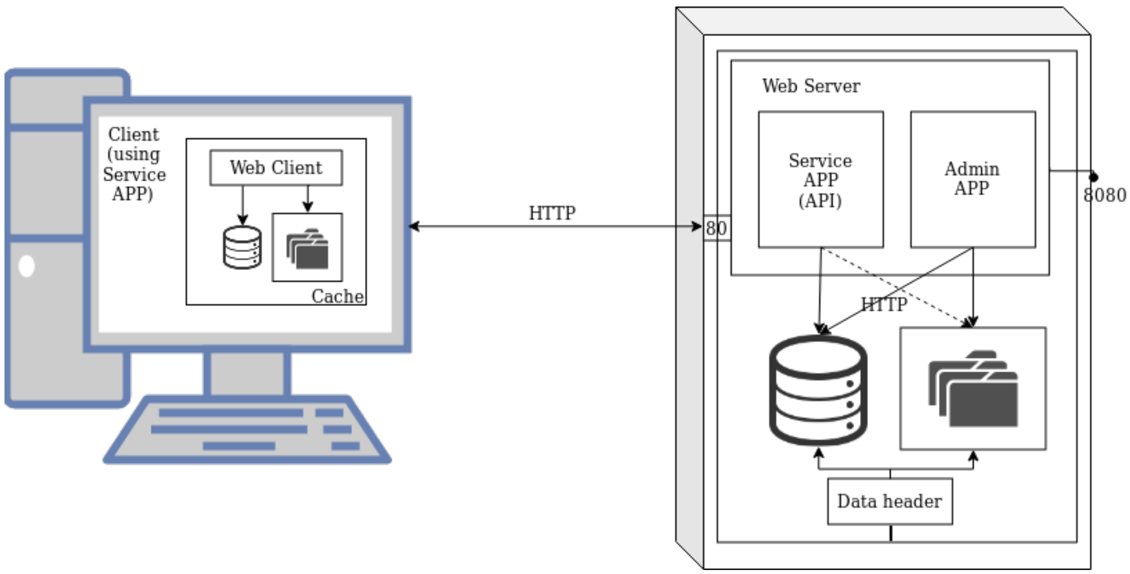


Fig. 22. General Software Architecture of *WAVE*

As it can be seen in figure 23, the architectural implementation for the Tracker is robust and secured, where end-user may access the website and send requests to the back-end through HTTP requests. The back-end further communicates with the created TensorFlow pipeline, returning the output through an API endpoint. After this, the returned data is treated and depicted to the end-user in the front-end. The Tracker is implemented with the factory architectural pattern, therefore leaving the creation and manipulation for multiple instances and usages of such AI interpreter, allowing users to upload and use different trained models.

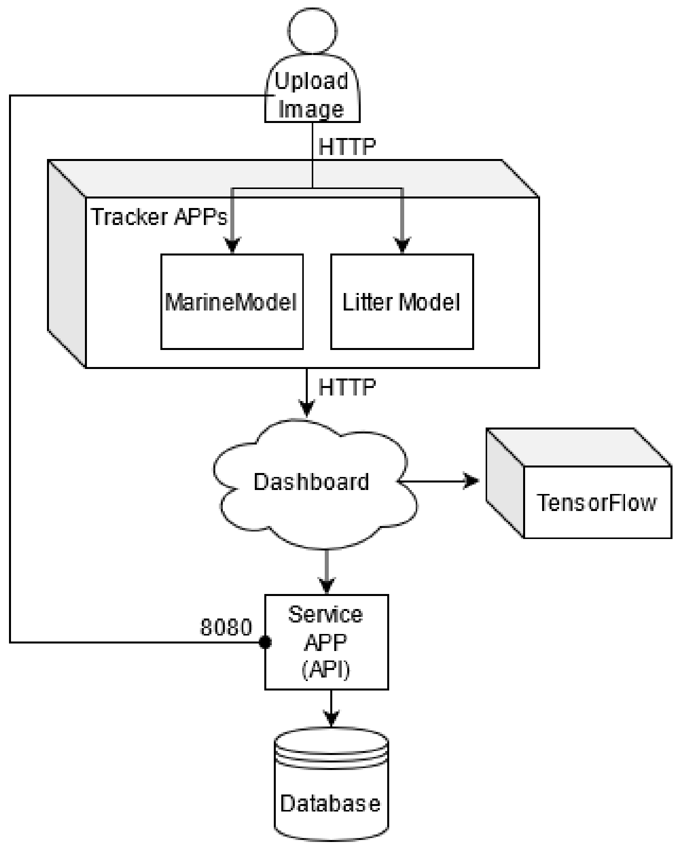


Fig. 23. Overall Tracker Architecture

3.2.1.3 Tracker Database

During the development of the system, future possible requirements were already thought and predicted resulting in the creation of modular tables inside of the database. In this dissertation, only tables refereed to the Tracker system are portrayed and described as seen in figure 25, further developing the Wave architecture. Following tables were created:

(i) **ai models** - This table is responsible for storing all of the existing model paths.
   (a) **id** - unique key;
   (b) **name** - commercial name of the generated model (e.g.: MarineMegafaunaNET, MarineLitterNET);
   (c) **path** - folder path to access the model (e.g.: /home/user/project/models/modelName/);

(ii) **ai labels** - This table is responsible to store all existing labels accross all used models.
   (a) **id** - unique key;
   (b) **name** - name of the label (e.g.: dolphin, turtle, whale, bird, seal);

(iii) **ai models has labels** - This pivot table is a result in a many to many (M2M) relations between the previous two described tables. In this table, each model is associated with their existing labels. (e.g.: one model can have numerous labels and a label can be referenced in numerous models, e.g. DolphinModel - label: dolphin, and in MarineModel - label: dolphin, shark , etc).

(a) **id** - unique key;

(b) **ai label id** - foreign key;

(c) **ai model id** - foreign key;

(iv) **ai detection images** - This table is responsible to store the image path and image metadata. It can be used to create new API endpoints, allowing to filter world coordinates into future heat-map usages (e.g. of flexibility).

(1) **id** - unique key;

(2) **path** - folder path to access the image;

(3) **width** - dimension of the image width;

(4) **height** - dimension of the image height;

(5) **latitude** - coordinate to represent latitude;

(6) **longitude** - coordinate to represent longitude;

(v) **ai detections** - Lastly, this table is responsible to associate the existing images with previously explained tables, therefore associating that image with the selected model, allowing the identification and types of labels, as well as coordinates of such identifications (bounding boxes), as seen in figure 24 below:

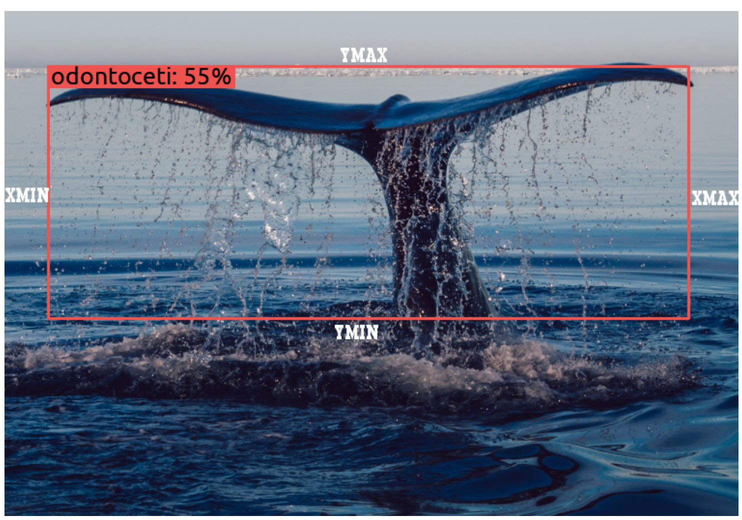


Fig. 24. Bounding Box Coordinates

(a) **id** - unique key;

(b) **user id** - foreign key;

(c) **ai detection image id** - foreign key;

(d) **ai label id** - foreign key;

(e) **ai model id** - foreign key;

(f) **accuracy** - result of the TensorFlow predicted precision;

(g) **xmin** - left coordinate of the bounding box;

(h) **xmax** - right coordinate of the bounding box;

(i) **ymin** - bottom coordinate of the bounding box;

(j) **ymax** - top coordinate of the bounding box;

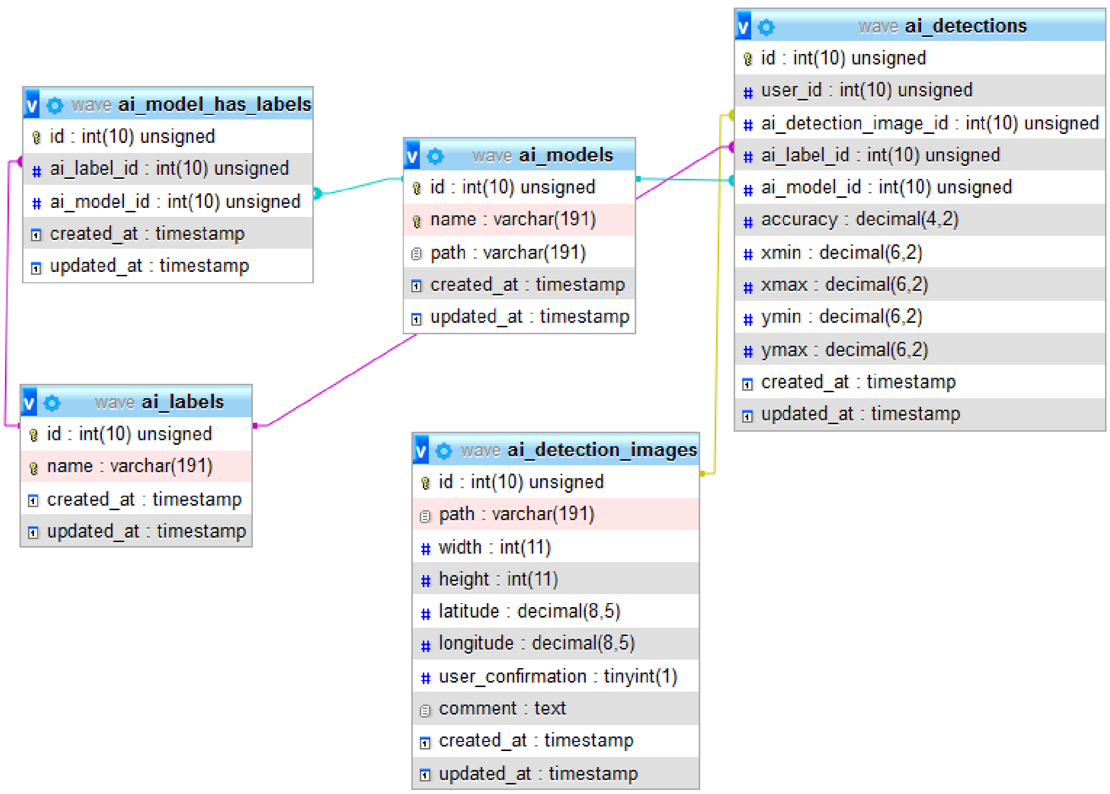


Fig. 25. Tracker Database Tables and Relations

3.2.1.4 User Interface

Once the back-end was placed into practice, Graphical UI was used for the data collection, interpretation and depiction. Some further insight in the individual components process and navigation is presented (see diagram in figure 26 explaining the usual procedure).

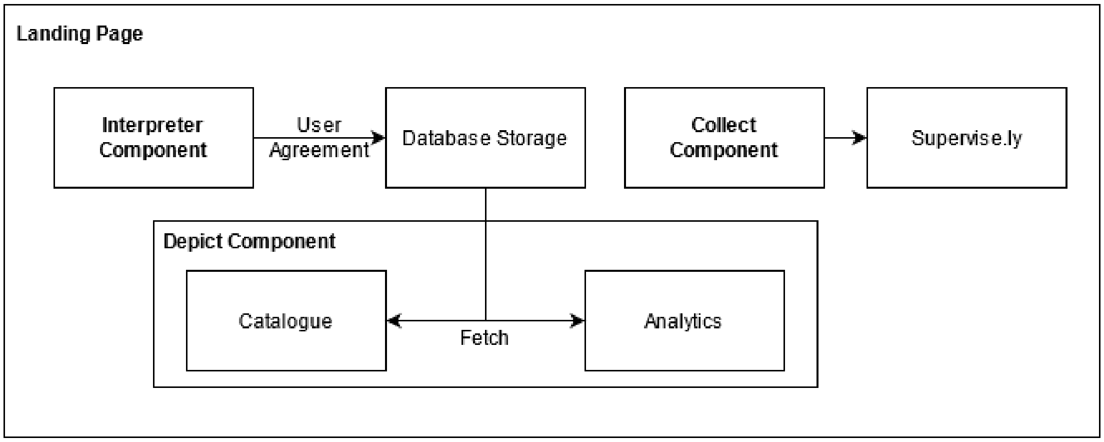


Fig. 26. Components interaction diagram

As aforementioned in section 3.2.1 and explained in the figure above, the developed component is implemented in *WAVE* dashboard and divided in three features seen in figure 27, representing the landing page: (i) Collection of imagery for analysis - upload function to the right; (ii) Tensorflow Interpreter - upload function to the left; and (iii) Depict - horizontal gallery bellow with statics graphics.

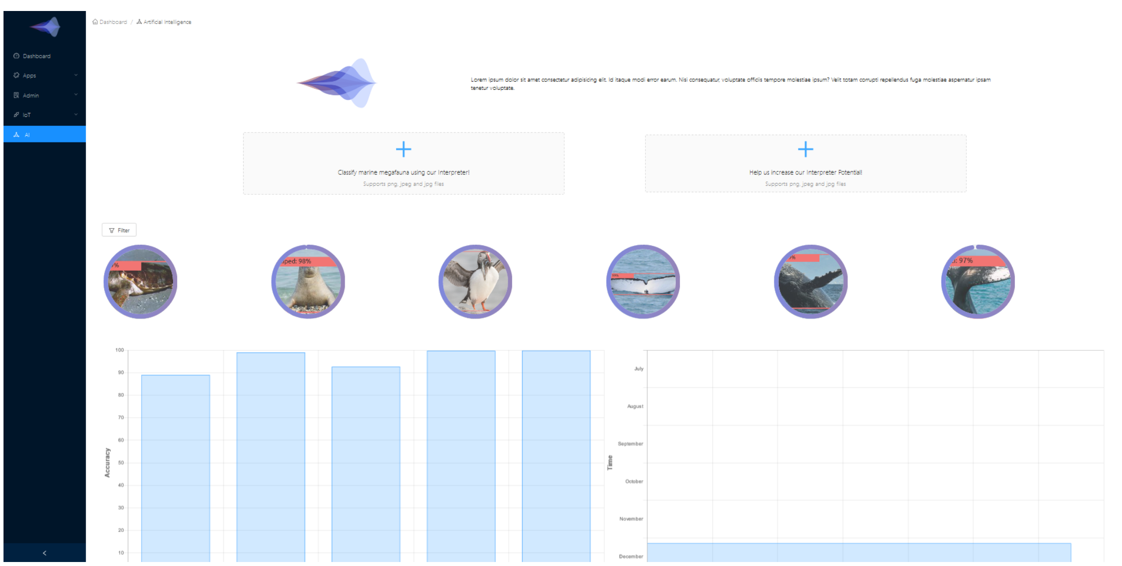


Fig. 27. *WAVE* Dashboard - Tracker Component

Following the established order of this dissertation, firstly the collection of imagery for future usage is made using the upload component to the right. This component consists of a API request for the third party website supervise.ly. The image is sent there and saved in a provisional dataset, figure 28, to be labeled and in a later stage cloned into the wider dataset.

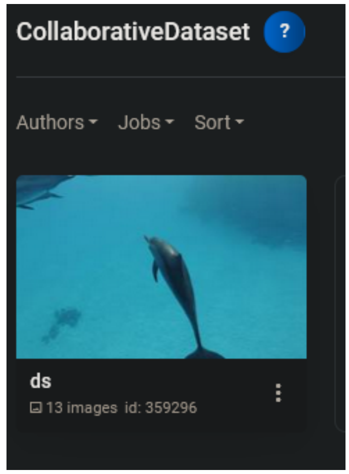


Fig. 28. Collaborative Dataset in the supervise.ly framework

Using an equivalent upload UI design, the interpreted component was created, however, communicating with the existing *WAVE* server, being responsible to test the image sent by the user against the created TensorFlow model. As aforementioned in section 3.2.1, the image is sent throughout the pipeline and returns through the API end-point the obtained accuracy. This result is presented in a pop-up modal with the average accuracy in the case that more than one detection is presented in that image. After this, the user chooses to agree and upload the image into the Tracker gallery or aborts the task. See figure 29 for more clarification.

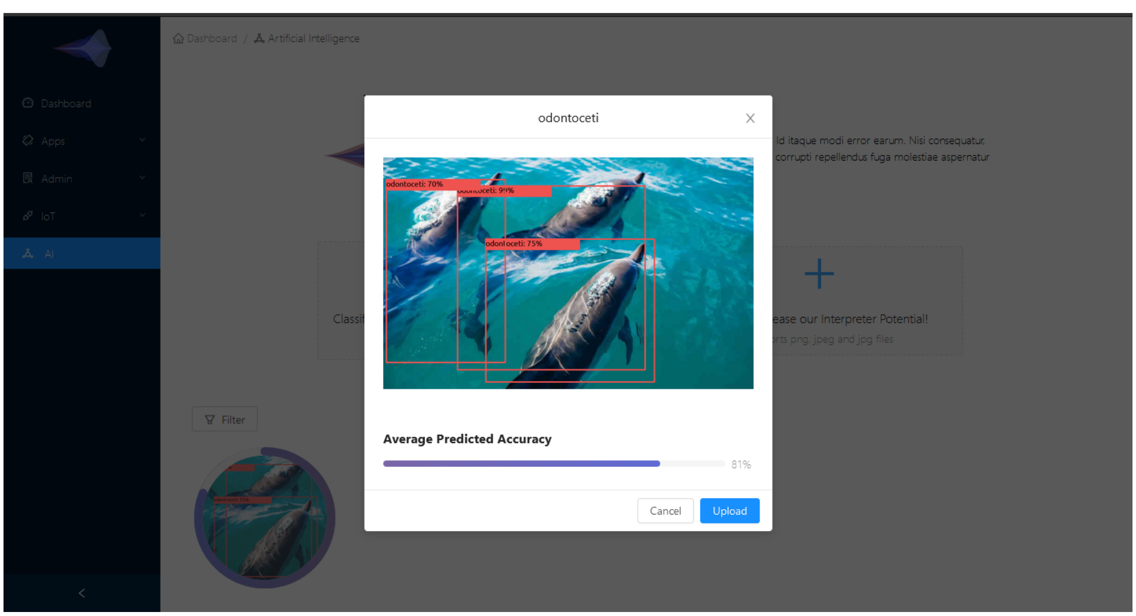


Fig. 29. Tensorflow Output Results

Following the same order, the depict implementation is split into two distinct components: (i) Gallery; and (ii) Statistics. The first component of the depict implementation, as seen in figure 27, the gallery is displayed in the horizontal view with a carousel format, where the last uploaded image is the top left one. However, querying filters were made to aid the user to showcase only the desired taxon groups (e.g. as in figure 30).

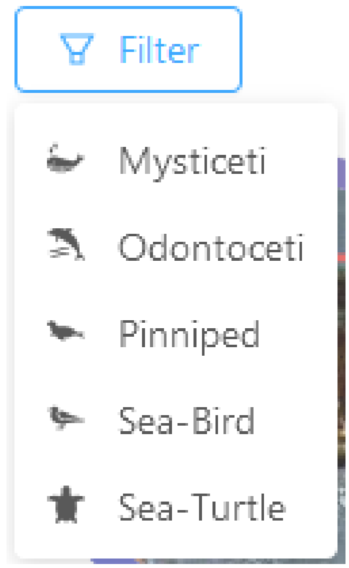


Fig. 30. Tracker Filters

Similarly, the explanation of the statistics is the next feature being presented. From an end-user point of view, representing the collected data may be important to showcase in this project, therefore graphics with statistics were designed using D3.js and react framework called react-chartjs-2[40]. This framework can be easily obtained using the npm (nodejs packaging manager) and the usage of it is in general straightforward. The data is retrieved by the developed API endpoint and sent to the component through the react props. The framework receives as argument X and Y axis respective names and values. As aforementioned in section 3.2.1, the implemented graphs were the following: (i) Accuracy per Species, figure 31 - returning the sum of each individual species' accuracy, divided by the number of pictures of that individual taxa (see equation 1); and (ii) Tracker time usage - represented by the number of validated images using the Tracker interpreter throughout the last six months, as seen in figure 32.

$$AverageAccuracy = \frac{\sum SpeciesAccuracy}{SpeciesOccurrence} \tag{1}$$

[40]https://www.npmjs.com/package/react-chartjs-2

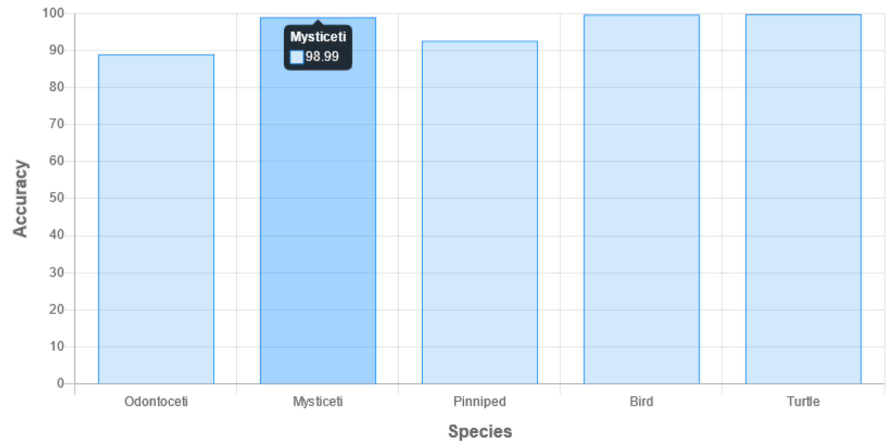


Fig. 31. Tracker Statistics - Average Accuracy per Species

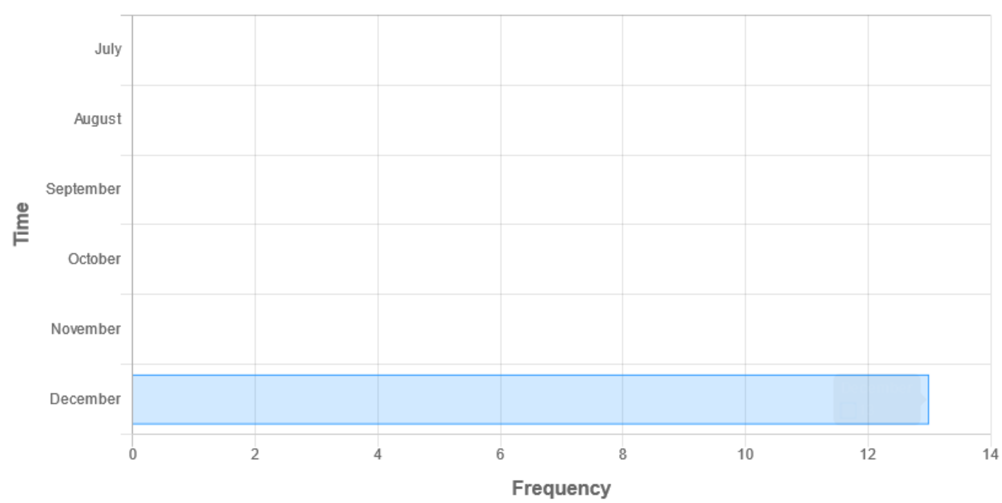


Fig. 32. Tracker Statistics - Average Usage in the last 6 Months

3.3 Study Setup

Once the Tracker system was successfully implemented, this dissertation document further describes the location of the study, means for data inquiry, including the procedures for model validation.

3.3.1 Location of the Study and Sample Background

The study focus was on Madeira[41] island, known for rich biodiversity and marine megafauna occurrence. Collected imagery was obtained from online sources, divided into 5 categories (whales, dolphins, seals, turtles and seabirds). For human validation of the Tracker system, System Usability Scale (SUS) test was taken by the experts of marine megafauna, and computer engineers. Two of the users were marine ecology experts in photo-id of dorsal fins and flukes. One of them has a PhD in biodiversity and ecosystems and is a supervisor of marine papers, being integrated into more than 15 international projects. The other candidate was a researcher and field biologist, working with the cetaceans. Moreover, the second candidate worked in the aforementioned project of section 2.2, MONICET. Furthermore, additional marine biologist student of the same area was invited to participate. As aforementioned, the rest (9) of the candidates were students without any marine background, with advance knowledge in computer engineering. The procedure of the study is further explained in section 4.2.

3.3.2 Study Protocol

In further, this dissertation document provides an insight into the performed experiments, outlining the data inquiry for both machine and user testing of the Tracker system.

(i) **[Q1]. Machine Testing: How accurate is the current trained model?**

(1) Previously trained models of marine megafauna (turtle, whale, dolphin, seal and sea bird) are benchmarked against different iterations.

(2) Each model is benchmarked using 20 images for each species category.

(3) Obtained statistics were collected.

In this step, previously trained model iterations were compared against each other (from 25k steps iteration to 150k steps iteration, in intervals of 25k). Furthermore, 20 images per specie (which were not used throughout the training phase), were further used to challenge the model accuracy. To compare each individual model, an output file with the training results is analyzed. This file is created and updated in the middle of the training for each successful epoch[42]. Each time an epoch was completed, the 1% of the images that were chosen randomly by the AI for later inference and bench-marking were used and saved these results in the output file. To aid in gathering and displaying all such obtained data and comparing such models, a neural

[41]Madeira Island Location and Climate: The Madeira Archipelago is located in the Atlantic Ocean, in the African plate, 978 km south of Portugal, approximately 33 degrees north latitude, about 700 km west of the African coast, almost the same latitude Casablanca and 450 km north of the Canary Islands, and has a total area of 796 km^2 [48]

[42]defined by the number of iterations needed for the learning algorithm to pass through the entire training dataset(number of images)

network framework of TensorFlow was used, called TensorBoard[43]. TensorBoard provides the visualization and tooling needed for machine learning, providing insight into multiple fields. Due to this framework's capabilities of running existing TensorFlow functions to display and return data, it made the process of depicting each iteration of the models quite straightforward. To analyze each model performance, precision, recall, loss F1 score and accuracy were reported.

(ii) **[Q2]. User Testing: How is the system perceived by the end-users?**

(1) In a web-page AI (Interpret), experts use the web interface to upload the newly acquired images from the field: (i) whole body, aerial or side imagery; (ii) dorsal fin, side imagery; or (iii) fluke, side imagery;

(2) Popup in GUI appears with the prediction accuracy and confidence rate output from TensorFlow;

(3) Person chooses to agree or not with the prediction and proceeds to upload the image to the gallery (publicly accessible on the page) or to keep it for themselves (image is not stored locally);

(4) Person navigates the gallery and explores the previous occurrence of such species;

(5) SUS test was performed on the studied sample.

More specifically, users were invited to open the page with the classifier. Next, they were asked to upload one image per marine megafauna (whale, turtle, seabird, seal, dolphin). AI returned the predicted accuracy. The user is invited to upload the imagery if it agrees with the classification, or to cancel in the opposite case. Once this is performed, the user is invited to filter the uploaded species and lastly, to explore the average accuracy of each uploaded image. System Usability Scale (SUS) was given to the user and demographics data (age, gender) were obtained, and familiarity of the usage of technology on a Like-rt scale 1-5. Users were the marine ecology experts, scholars applying the Photo-id of dorsal fins and flukes, and potential future stakeholders. The sample group of 12 persons had age rate between 23 and 44 years old. Participants validated the implemented UI and the trained model.

[43]https://www.tensorflow.org/tensorboard

4 Results

In this chapter, the results of section 3.3 are portrayed when a model is trained, validated and tested, depicting the model's precision, recall, loss, F1 score and accuracy. However as mentioned in section 3.1.4 the current used repository only allows for number of iteration steps and selected model change. Therefore, for iterate the model multiple instances of same dataset were used in the training to generate new models changing internal variable (number of training steps). As for selected model variable, the only options available were mobilenet and mobilenet quantized, since quantized models are used for IoT devices this setting was skipped. Moreover, results from the System Usability Scale (SUS) are reported below.

4.1 Machine Testing: Model Accuracy

(i) **Precision**

The precision of the model answers how much a prediction of the positive identifications was actually correct. These values range between 0.0 and 1.0 and can be understood as how much a model is correct (e.g: a precision of 0.6 means that a model is correct 60% of the time). This value comes from a fraction of the True Positives identified in an image divided by the true positives plus false positives, as seen in equation 2. Since most of the used imagery in the training had a bounding box that occupies more than 40% of the image, precision at 50% intersection of union was used. This means that the drawn bounding box by the inference occupies at least 50% of the same bounding box drawn in the dataset creation. An example is seen in the figure 34. As a result, it is possible to compare when using images of the same type how accurate can the model be. Returned accuracy for each model iteration is the following, as seen in figure 33.

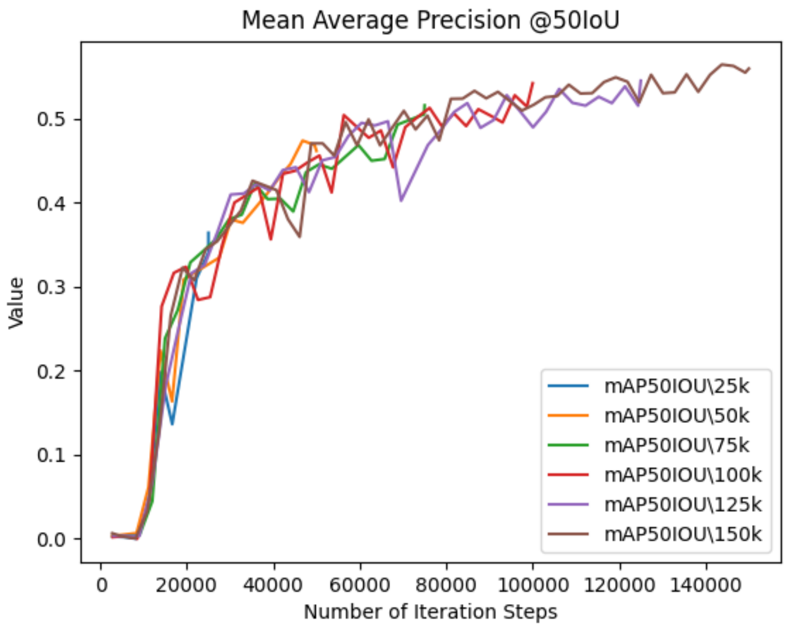


Fig. 33. Mean Average Precision @ 50IoU

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{2}$$

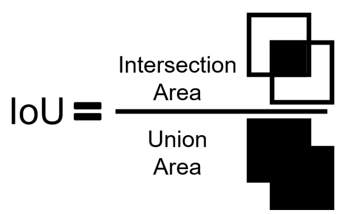


Fig. 34. Intersection of Union Equation with example

(ii) **Recall**

Another important value to analyze is the recall. This value is represented by how much of the actual positives were identified as being the correct ones. Equation 3 describes the ratio between true positives and false negatives.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{3}$$

In other words, this value represents the number of correct answers divided by the number of the actual correct answers that should have been returned (e.g: 0.35 recall value means that 35% of the time the model returns the expected answer). As a result, the recall of each model iteration can be comprehended below, see figure 35.

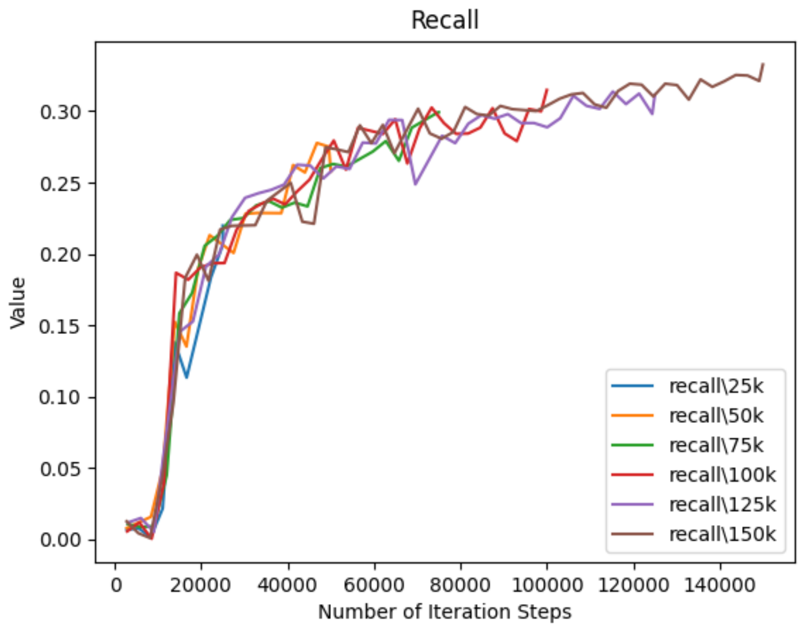


Fig. 35. Recall

(iii) **Loss**

When a model is being developed, it is important to minimize the error, therefore maximizing the productivity of a system. Therefore, while in the training the algorithm runs multiple regressions functions, such loss function corresponds in our case to the mean square error (MSE) function. This function is calculated as the average of the squared difference between the predicted value and the actual target value, as seen in the equation 4.

$$Loss = \frac{1}{n}\sum_{i=1}^{n}(Target - Predicted)^2 \tag{4}$$

The result of such calculation is always positive regardless of the predicted and target values. However, the closer to 0, the minimized the loss. Therefore, this is an important aspect, representing the capability of future improvements with new iterations. The results for each iteration can be analyzed below, in figure 36.

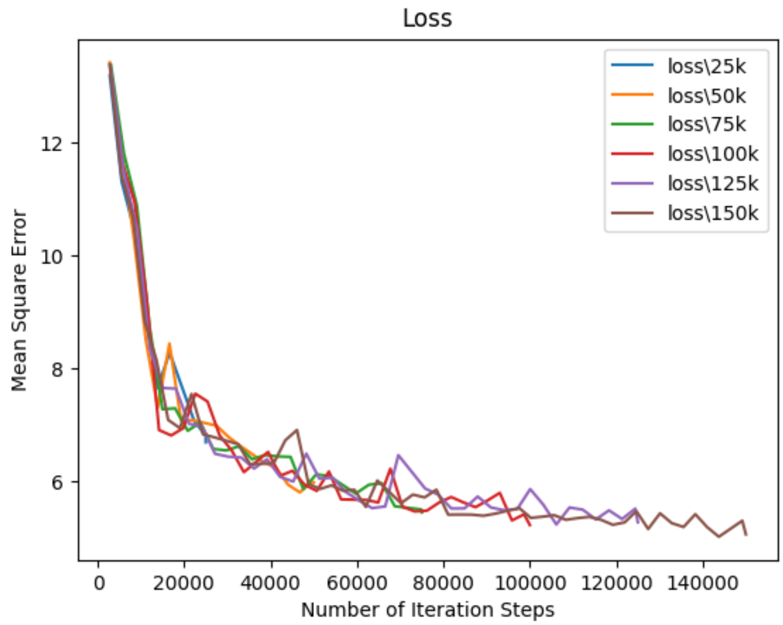


Fig. 36. Loss

(iv) **F1 Score**

Among other calculation, the obtained F1 score is a measure of the ratio between the precision and recall. This value is needed to find the correct balance between precision and recall. To calculate such balance, it is necessary to compute such equation:

$$F1Score = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

Thus, the F1 Score is an adequate choice to compare different models predicting the same imagery. This value ranges between 0 and 1, where the higher, the better. However, this value depends heavily on how imbalanced a dataset is (e.g: predicting wrong species in five results in a lower F1 score than if it had fewer classes). F1 score is depicted in figure 37.

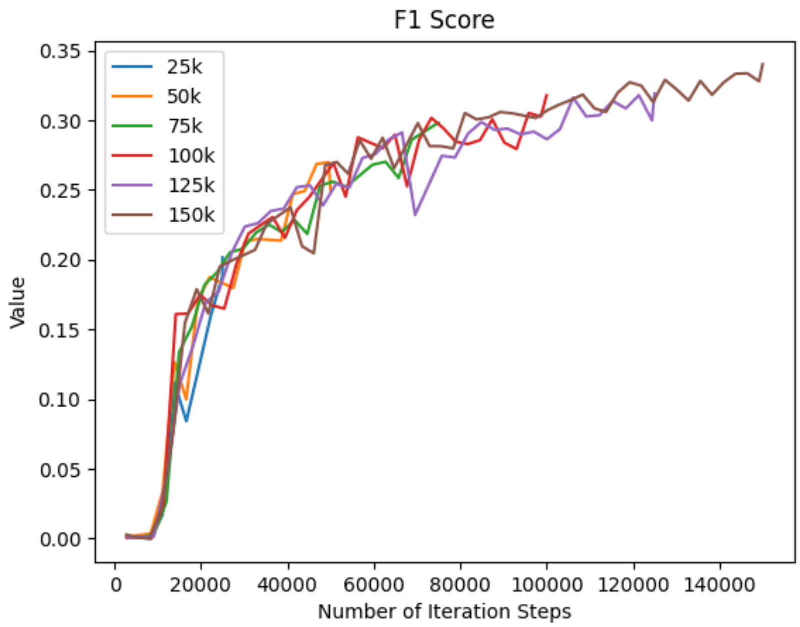


Fig. 37. F1 Score

(v) **Precision**

Lastly, after analyzing and depicting the results of the evaluation of training the models, an inference tests with 20 images of each species category chosen randomly was performed by an individual guest, using the google search engine to collect the imagery. The results depict the precision returned by the model in each image, however for categorization purposes of models, it is called as an accuracy of each model. Each model was tested with the imagery and the obtained results are depicted in the figure 38.

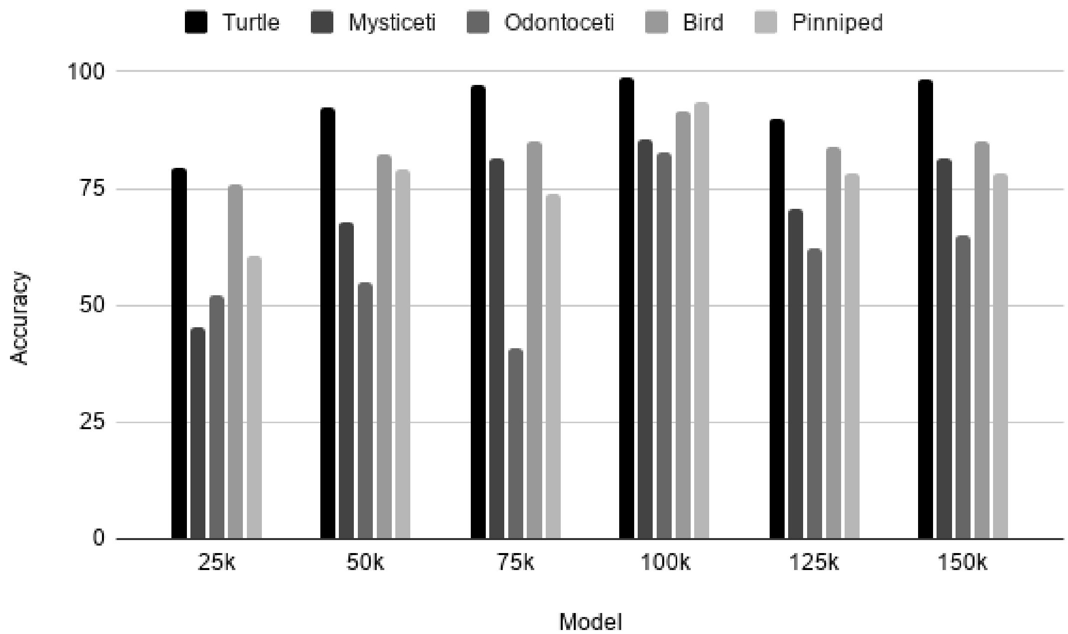


Fig. 38. Model Iteration Comparison

As it can be understood in the aforementioned figure, each iteration usually produces a better average accuracy, however from the 100k model to the others, the average accuracy actually declined and then started to rise again. Looking individually at the detection, it was found that the 100k model detects less objects, therefore having a higher accuracy, as seen in the figure 39.

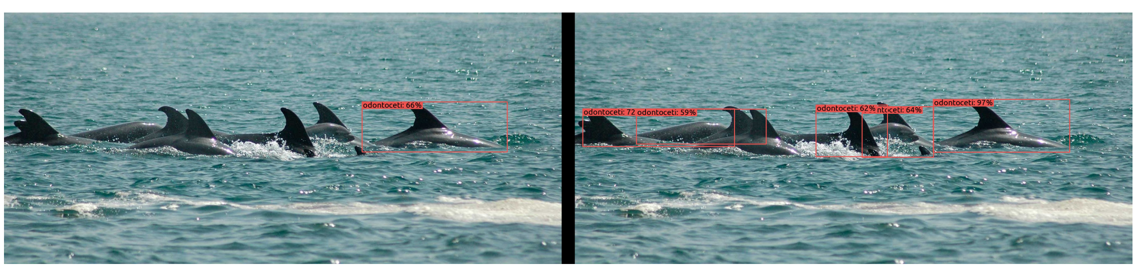


Fig. 39. Detected dolphins using the 150k iterations

All things considered, the model used for the Tensorflow pipeline and identification of species chosen was the 150k, because of the capability of detecting multiple marine megafauna more precisely. In below, accuracy per species category as well as grand average the values is depicted in figure 40. This model was further used for user study validation.

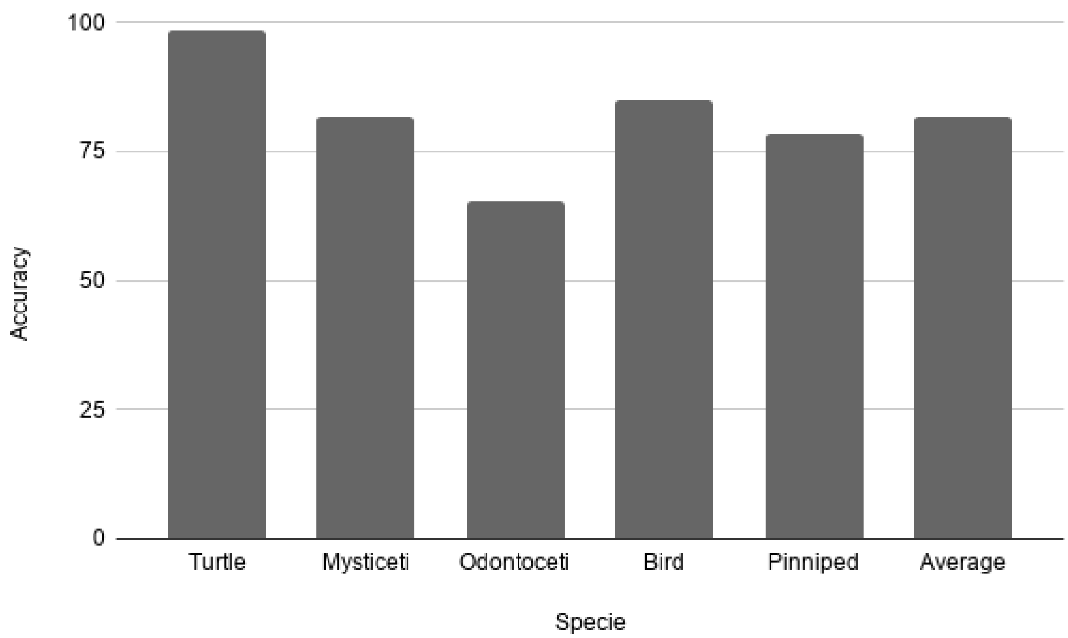


Fig. 40. Marine Megafauna 150k Model

4.2 User Testing: Dashboard Evaluation

Sample size of 12 participants validated the proposed system with the applied 150k model, providing the feedback about the usage of such a system. Obtained SUS (Using standard Likert-type scale, where the response scales use anchors such as 1=Strongly Disagree, 2=Disagree, 3=Neutral, 4=Agree, 5=Strongly Agree) results are portrayed bellow, table 6:

1. **Familiarity with technology**

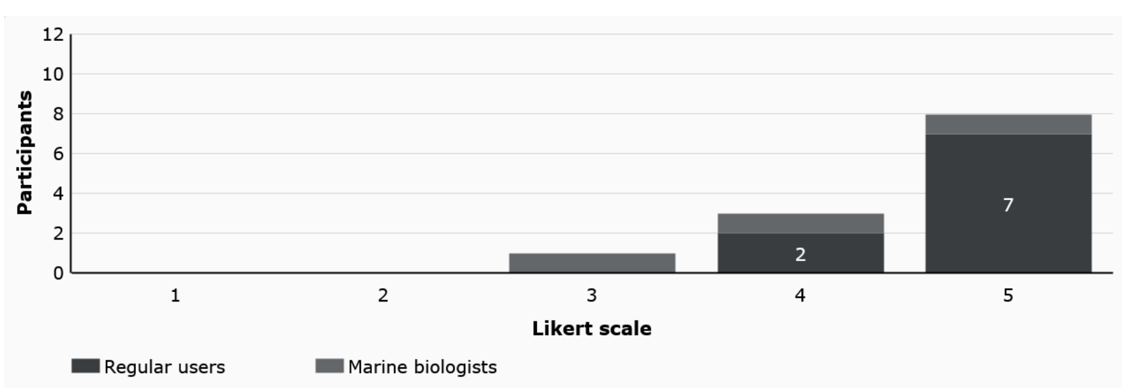


Fig. 41. Question 1 - I am familiar with Technology

Table 6. System Usability Scale Results and Score

	1	**2**	**3**	**4**	**5**
I found the system unnecessary complex	11	1	0	0	0
I would like to use this system frequently	0	3	4	5	0
I thought the system was easy to use	0	0	0	1	11
I think that I would need the support of a technical person to be able to use this system	10	0	1	1	0
I found the various functions in this system were well integrated	1	0	0	4	7
I thought there was too much inconsistency in this system	8	2	2	0	0
I would imagine that most people would learn to use this system very quickly	0	0	0	4	8
I found the system very cumbersome to use	5	6	0	0	1
I felt very confident using the system	0	0	1	4	7
I needed to learn a lot of things before I could get going with this system	5	4	2	1	0
SUS Score	**74,17**				

5 Discussion

In this section, explanation of the results presented at previous section 4 will be interpreted. As well as explaining the full contribution of this dissertation. Moreover, all the constraints encountered in the process will be presented and some insight into future decisions and works.

5.1 Experimental Findings

As aforementioned, this section describes all previously presented results, section 4, and depicts in the following research questions presented in section 3.3.2

(i) **[Q1]. Machine Testing: How accurate is the current trained model?**
Looking merely in the returned global accuracy of figure 40, the current model presents an average prediction accuracy of 81.72%, however as previously explained, the values only indicate the average of successful identification. To understand how accurate is the current model, it was necessary look at the model's precision function. Therefore, current model true precision (mAP@50IoU) is 0.55, this values means that current model predicts accurately 55% of the time, drawing the bounding box the way expected. This value can be comprehended as low model performance. However, in reality, since we are using the intersection of the union to return a more precise result, in fact, 0.55 is a very good average for a model. As for an increase and upgrade of the current model we should look at loss function. The presented model has a loss (MSE) of 5.06. As explained previously, these values indicate a way of minimizing the error, therefore, when as close to 0 the better. With this in mind, and looking at the current loss graph, figure 36 the model requires to be trained with a wider and more varied collection of imagery.

(ii) **[Q2]. User Testing: How is the system perceived by the end-users?**
The implemented UI was validated in a user study where the results were presented in section 4.2. As depicted in these results, most of the users were familiarized with technology averaging a 4,7 on a scale of 1-5. As a result, the average for usage, learning curve and expertise required were pretty straightforward values, because the implemented system had an average of the inconsistency of 1,5 and the users rate the integration of all the implemented functions with an average score of 4,6. Obtained SUS had a score of 70,84 and 75,28, for the marine experts and regular users, respectively.

5.2 Research Contributions

This section connects the implemented work of this dissertation with the research contributions presented in section 1.2. In this section, previous questions will be answered and explained.

(i) **[RQ1]. How a software can be used to decipher and depict the marine megafauna presence?**
The current dissertation document provides a web service where any individual can identify marine megafauna encounters, allowing non novice users to participate. the system provides the current identifications of species, which are saved and presented in *WAVE* dashboard into a image gallery. Further statistical data is also presented, outlining the model performance and interest in classification of specific marine taxa.

(ii) **[RQ2]. How well does the image vision perform when estimating marine megafauna?**
At a first glance, looking solely at the results of image vision performance seems to be consistent. However, during the making of this dissertation and off-task testing, surprising results were encountered, where the author was not sure about the species and found out that image vision did not actually perform an error, and instead taught the author the correct answer (e.g. Imagery of Toothless whales - odontoceti where only the fluke or dorsal fins were present in the image - such predictions were correct). This is also consistent with the literature review and used software, providing the alternative in using the object detection against the trailing edge contours. Presented image vision algorithm also performed well in a large data corpora data environment, removing the time consuming need for human manual inspection.

(iii) **[RQ3]. Which is the feedback from marine biologists and other stakeholders when using this system?**
Marine biologists obtained the same SUS as the other group, After completing the SUS test they offered suggestions, improvements and expressed an overall interest in the system. Firstly, the results of the marine biologists show interest in such a system, averaging 3,3. This result seems far low from the expected, however, the marine biologists after taking the test mentioned that as a starting point, it is a really good concept. Indeed, these experts pointed out that to them the requirements of such a system are still far from the expected. The marine biologists described that having a catalogue is important but not for the marine taxa. Moreover, we learned that for marine megafauna experts the importance of collecting data to analyze later is significant, and having an automated system or semi-automated system would provide a big step, reducing the amount of time(human resources). However, for the identification, marine experts pointed out the need for such a tool to help identify individual species, instead of groups. In addition, they also pointed out that different experts go on different trips and some

data can get noisy, and for this reason, the provided Tracker system based on deep learning can be a good integration for their system.

5.3 Literature Analysis

In this section current developed system is compared with existing methodologies described in section 2.1. Therefore, to compare each develop component added in *WAVE* dashboard to the technologies previously explained the same order will be used: **(i)** Collect; **(ii)** Interpret; and **(iii)** Depict.

(i) **Collect**

As previously explained, current technologies and techniques for assessing megafauna imagery rely upon land's tools (whale watcher), sea trips (marine biologist field trips), and space (ESA SPACEWHALE Project). The component implemented for the collection of imagery does not try to solve this problem, but to aid with a way to store collected data in a single source (*WAVE* dashboard). However, the currently implemented model can be used in IoT devices in sea-vessels and collecting continuous imagery.

(ii) **Interpret**

As described and focused in SoA, image vision and software already exist. However, since the focus of this dissertation was to build something from scratch and validate it against existing technologies such as the depicted in section 2.2.2 table 3. The current development component consists of an endpoint calling a Tensorflow pipeline that uses the newly created ML model. *WAVE* dashboard interpreter follows a similar system type: Semi-automated, requiring the usage of the endpoint to predict and interpret desired imagery. However, matching focus is not locked for dorsal fins only as others existing tools. The developed model is capable of identifying megafauna with full-body, dorsal fin, fluke, or top view underwater species, see image 42. As for response time required to identify and depict megafauna in the imagery implemented solution takes less than ten seconds for each image.

Fig. 42. Example of a partial top view underwater imagery

(iii) **Depict**

As for the technologies presented in section 2.2.3, *WAVE* dashboard and the implemented components of this dissertation take the same approach as Monicet[55] platform. Putting current developed solution versus the presented by Monicet[55] both are a dashboard focus of collecting and depicting data. As previously described, Monicet[55] work with marine biologists to verify and catalog marine megafauna species manually. *WAVE* dashboard, on the other hand, does not have aid from marine scientists to collect, upload and identify. The presented solution is a semi-automated system that requires any user to upload imagery to help build a catalog. Therefore, each dashboard has its pros and cons, Monicet[55] is built for marine biologists to aid them to store and catalog local species manually, wheres *WAVE* focus on receiving any type of crowd-sourcing to build their catalog with the aid of a developed AI Model.

5.4 Constraints and Future Work

This section provides insight into the dissertation constraints and choices through all the implementation process, as well as future improvements.

Five marine megafauna classes were used, while for the estimation of overall surface marine biodiversity, more taxa should be incorporated. Regarding the created pipeline, GPU used actually depended on the available resources from the Google Collab, thus suggesting further training steps at the proprietary GPUs where possible. Detecting the individuals remains as a challenge.

Still, this dissertation contributes to the development of a semi-automated system where the user uploads and agree/disagree with the output from the deep learning algorithm. This was the first step in a requirement to fulfill the needs of marine biologists. The experts marine biologists pointed out (as aforementioned in section 5.2), that a proposed dashboard can help maintain the consistency between trips with a different marine biologist. However, the implemented system would need a more practical use, differentiating between individuals, analyzing the coloration of their dorsal fins and flukes, and comparing with the existing individuals already encountered in previous trips.

To fulfill all aforementioned requirements more time would be required to gather enough imagery of individuals or species as well as a new dataset and therefore to train a new AI model. Since the implementation of the UI was designed using the described *WAVE* architecture, the only requirement to change will be to generate the new models and adding to the Wave database the model path for future calls in the implemented web-based TensorFlow pipeline.

Another concern from a developer point of view is the time and human resources required to extend and create the new datasets, therefore models. While all the gathered imagery was manually annotated, future works can try to resolve this. As aforementioned in this dissertation, current machine learning can be split in supervised learning (manually labelled data) and unsupervised learning (to be labelled by the AI). Therefore, in a future work the collect component can be reworked in two ways: (i) Supervised Learning - where the system asks the user input (to draw the square and label) and send the annotation throughout *supervise.ly* API; or (ii) Unsupervised Learning - use current developed model to identify and, in another pipeline, connect the output data generating the required files to upload to *supervise.ly*.

6 Conclusion

After designing and testing the implemented system throughout this dissertation, it is possible to assert that the proposed dashboard for depicting marine megafauna may provide a significant impact not just to the marine ecology but also to the overall IT industry, where the algorithms may be tailored to identify specific niches, necessary to discriminate the individual species, occurring at the multiple locations. Such can provide the analysis of the migration flows, where the proposed AI platform can be a more robust tool in observing the routes of marine biodiversity.

Clearly, the entire field of marine megafauna experts is expanding, as well as the requirements and conditions that these ecological experts may need. With the obtained feedback from the users validating this system, it is possible to use the proposed system, which can reduce the amount of time, when marine ecologists analyze the obtained footage from the field work. For instance, marine biologists can upload the obtained image dataset once they are back from the field work and await for the proposed model to yield the accuracy in detecting individual species at multiple locations.

Indeed, the proposed dashboard offers significant value to the field of marine ecology and aims to provide the continuously evolving AI, supporting the typical usage of photo-identification. Conversely, proposed apparatus does not intend to remove the need of the marine ecologists, but rather to provide an extension tool for speeding up such identification process.

Finally, as envisioned throughout this dissertation, a machine learning model can be applied in multiple environments that have an adequate computing power. Such may range from ROVs, AUVs, UAVs, USVs, IoT devices to simple web pages served by a web server. As aforementioned, this dissertation provided an accessible tool for gathering pictures, identifying the taxa, and display the data in a dashboard. Proposed tool is publicly available on a web-page of *WAVE* lab team[44], serving to impact the UN SDG 14: Life Below Water, by providing the tools for increasing of marine literacy and reducing the marine litter.

[44] `http://wave.arditi.pt`

References

[1] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: A System for Large-Scale Machine Learning. In: 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16). pp. 265–283. {USENIX} Association, Savannah, GA (2016), `https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi`

[2] Adams, J.D.: FinBase World Marine Mammal Science Conference 2019 Office of Protected Resources p. 14 (2019)

[3] Al-hajj, S., Pike, I., Fisher, B.: Interactive Dashboards : Using Visual Analytics for knowledge Transfer and Decision Support (2010)

[4] Alessandrini, A.: Photo-identification Catalogue of Globicephala macrorhynchus (2017)

[5] Allen, J.B., Moore, R.T.: CurvRank Background - Testing - Implementation (December) (2019)

[6] AudioMoth: AudioMoth | openacousticdevices (2018), `https://www.openacousticdevices.info/audiomoth`

[7] Basheer, I., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and application. Journal of Microbiological Methods **43**(1), 3–31 (dec 2000). https://doi.org/10.1016/S0167-7012(00)00201-3, `https://linkinghub.elsevier.com/retrieve/pii/S0167701200002013`

[8] Brief, R.: Marine Biodiversity: What is it and why it is so important? Reef Briefs, All About the Belize Barrier Reef, `https://ambergriscaye.com/reefbriefs/briefs77.html`

[9] Canon: Canon 18x50 IS All Weather - Image Stabilisation Binoculars - Canon Portugal, `https://www.canon.pt/for{_}home/product{_}finder/binoculars/optical{_}image{_}stabilisation/18x50{_}is/specification.html`

[10] CDV: Pedro Miguel Cruz « CDV Lab (2019), `https://cdv.dei.uc.pt/authors/pedro-cruz/`

[11] Cetalingua: Homepage - Cetalingua Project, `https://cetalingua.com/`

[12] Chen, A.: Pattern Radio: Whale Songs (2019), `https://patternradio.withgoogle.com/`

[13] CSSAuthor: 100+ JavaScript Frameworks For Web Developers (2019), https://cssauthor.com/javascript-frameworks/

[14] Cush, C.: Gulf of Mexico Dolphin Identification System (GoMDIS) p. 15

[15] Darwin: DARWIN: photo-identification, http://darwin.eckerd.edu/?page=photo{_}identification.html

[16] DeMares, R.: Human Peak Experience Triggered by Encounters with Cetaceans. Anthrozoös **13**(2), 89–103 (jun 2000). https://doi.org/10.2752/089279300786999914, https://www.tandfonline.com/doi/full/10.2752/089279300786999914

[17] DensityDesign: DensityDesign Lab | RAWGraphs, https://densitydesign.org/projects/https://densitydesign.org/research/raw/

[18] DigitalGlobe: DigitalGlobe | WorldView-3, http://worldview3.digitalglobe.com/

[19] EGuirado: EGuirado/CNN-Whales-from-Space: Detecting and counting Whales with convolutional neural network from Google Earth imagery (2019), https://github.com/EGuirado/CNN-Whales-from-Space

[20] ESA: SPACEWHALE - MAPPING WHALES FROM SPACE | ESA Business Applications (2018), https://business.esa.int/projects/spacewhale

[21] ESA: Newcomers Earth Observation Guide | ESA Business Applications (2019), https://business.esa.int/newcomers-earth-observation-guide

[22] Expert System Team: What is Machine Learning? A definition - Expert System (2017), https://www.expertsystem.com/machine-learning-definition/

[23] Few, S.: Information Dashboard Design. The effective visual communication of data Sebastopol (2006), http://www.mendeley.com/research/information-dashboard-design/

[24] Firefox: What is JavaScript? - Learn web development | MDN (2019), https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First{_}steps/What{_}is{_}JavaScripthttps://developer.mozilla.org/en-US/docs/Learn/JavaScript/First{_}steps/What{_}is{_}JavaScript{#}A{_}high-level{_}definition

[25] Flukebook: Flukebook, https://www.flukebook.org/

[26] Funchal, M.: Tariff – Marina do Funchal (2019), https://www.marinadofunchal.pt/en/tariff/

[27] Gasc, A., Sueur, J., Jiguet, F., Devictor, V., Grandcolas, P., Burrow, C., Depraetere, M., Pavoine, S.: Assessing biodiversity with sound: Do acoustic diversity indices reflect phylogenetic and functional diversities of bird communities? Ecological Indicators **25**, 279–287 (feb 2013). https://doi.org/10.1016/j.ecolind.2012.10.009, http://dx.doi.org/10.1016/j.ecolind.2012.10.009https://linkinghub.elsevier.com/retrieve/pii/S1470160X12003603

[28] Ghosh, A., Kumar, H., Sastry, P.S.: Robust Loss Functions under Label Noise for Deep Neural Networks. 31st AAAI Conference on Artificial Intelligence, AAAI 2017 pp. 1919–1925 (dec 2017), http://arxiv.org/abs/1712.09482

[29] Gilman, A., Pawley, M.D.M., Hupman, K.E., Leist, A., Stockin, K.A.: Photo-ID Ninja (2019)

[30] Google: Detecting Faces | Google Cloud Vision API Documentation | Google Cloud Platform (2019), https://cloud.google.com/vision/quotashttps://cloud.google.com/vision/docs/detecting-faces

[31] Google LLC: Vision AI | Derive Image Insights via ML | Google Cloud (2019), https://cloud.google.com/vision/

[32] Gray, J.S.: Marine biodiversity: Patterns, threats and conservation needs. Biodiversity and Conservation **6**(1), 153–175 (1997). https://doi.org/10.1023/A:1018335901847, https://doi.org/10.1023/A:1018335901847

[33] Guirado, E., Tabik, S., Alcaraz-Segura, D.: Automatic whale counting in satellite images with deep learning ECOPOTENTIAL-improving future ecosystem benefits through earth observations View project Ecosystem and Socio-Ecosystem Functional Types (E&SEFT Project): Integrating biophysical and social fu. bioRxiv p. 443671 (2018). https://doi.org/10.1101/443671, https://www.biorxiv.org/content/early/2018/10/16/443671

[34] Guirado, E., Tabik, S., Rivas, M.L., Alcaraz-Segura, D., Herrera, F.: Whale counting in satellite and aerial images with deep learning. Scientific Reports **9**(1), 14259 (dec 2019). https://doi.org/10.1038/s41598-019-50795-9, http://www.nature.com/articles/s41598-019-50795-9

[35] Haimeh: AppUsage · haimeh/finFindR Wiki (2018), https://github.com/haimeh/finFindR/wiki/AppUsage

[36] Happywhale: Happywhale (2015), https://happywhale.com/home

[37] HighCharts: HighCharts Demos (2009), https://www.highcharts.com/demohttp://www.highcharts.com/demo

[38] Higino Faria: Vigia de cetáceos, uma nova profissão para ajudar a conservar os reis do oceano - Wilder (2019), https://www.wilder.pt/historias/vigia-de-cetaceos-uma-nova-profissao-para-ajudar-a-conservar-os-reis-do-oceano/

[39] Hill, A.P., Prince, P., Piña Covarrubias, E., Doncaster, C.P., Snaddon, J.L., Rogers, A.: AudioMoth: Evaluation of a smart open acoustic device for monitoring biodiversity and the environment. Methods in Ecology and Evolution **9**(5), 1199–1211 (may 2018). https://doi.org/10.1111/2041-210X.12955, http://doi.wiley.com/10.1111/2041-210X.12955

[40] Holzer, A., Ondrus, J.: Mobile application market: A developer's perspective. Telematics and Informatics **28**(1), 22–31 (feb 2011). https://doi.org/10.1016/j.tele.2010.05.006, https://linkinghub.elsevier.com/retrieve/pii/S0736585310000377

[41] Horn, M.S., Tobiasz, M., Shen, C.: Visualizing Biodiversity with Voronoi Treemaps. In: 2009 Sixth International Symposium on Voronoi Diagrams. pp. 265–270. IEEE (jun 2009). https://doi.org/10.1109/ISVD.2009.22, http://ieeexplore.ieee.org/document/5362305/

[42] Hoyt, E.: Whale Watching 2000 : Worldwide Tourism Numbers , Expenditures , and Expanding. No. 44 (2000)

[43] La Sorte, F.A., Lepczyk, C.A., Burnett, J.L., Hurlbert, A.H., Tingley, M.W., Zuckerberg, B.: Opportunities and challenges for big data ornithology. The Condor **120**(2), 414–426 (may 2018). https://doi.org/10.1650/CONDOR-17-206.1, https://academic.oup.com/condor/article/120/2/414-426/5153000

[44] Lacity, M., Willcocks, L., Feeny, D.: Commercializing the Back Office at Lloyds of London:. European Management Journal **22**(2), 127–140 (apr 2004). https://doi.org/10.1016/j.emj.2004.01.016, https://linkinghub.elsevier.com/retrieve/pii/S0263237304000258

[45] LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–44 (may 2015). https://doi.org/10.1038/nature14539, http://arxiv.org/abs/1807.07987http://www.nature.com/articles/nature14539http://www.ncbi.nlm.nih.gov/pubmed/26017442

[46] Leff, A., Rayfield, J.: Web-application development using the Model/View/Controller design pattern. In: Proceedings Fifth IEEE International Enterprise Distributed Object Computing Conference. pp. 118–127. IEEE Comput. Soc (2001). https://doi.org/10.1109/EDOC.2001.950428, http://ieeexplore.ieee.org/document/950428/

[47] Leung, H., Haykin, S.: The complex backpropagation algorithm. IEEE Transactions on Signal Processing **39**(9), 2101–2104 (1991). https://doi.org/10.1109/78.134446, http://ieeexplore.ieee.org/document/134446/

[48] Madeira Best: Madeira Island Location and Climate (2017), https://madeira.best/guide/facts-about/madeira-island-location-and-climate

[49] Maier, G., Schneider, F., Feldmann, A.: A First Look at Mobile Hand-Held Device Traffic. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 6032 LNCS, pp. 161–170 (2010). https://doi.org/10.1007/978-3-642-12334-4$_1$7, http://link.springer.com/10.1007/978-3-642-12334-4{_}17

[50] Mann, J.: Dolphin Matching Using Google Cloud Vision (DMUGCV) (2018)

[51] Marengoni, M., Stringhini, S.: Tutorial: Introdução à Visão Computacional usando OpenCV. Revista de Informática Teórica e Aplicada **16**(1), 125–160 (mar 2010). https://doi.org/10.22456/2175-2745.11477, https://seer.ufrgs.br/rita/article/view/rita{_}v16{_}n1{_}p125

[52] Matheson, C.: iNaturalist. Reference Reviews **28**(8), 36–38 (oct). https://doi.org/10.1108/RR-07-2014-0203, https://www.emerald.com/insight/content/doi/10.1108/RR-07-2014-0203/full/html

[53] McCandless, D.: Information Is Beautiful. Information is beautiful (June), 1–2 (2012), https://informationisbeautiful.net/http://www.informationisbeautiful.net/visualizations/20th-century-death/{%}5Cnhttps://docs.google.com/spreadsheet/ccc?key=0Aqe2P9sYhZ2ndDFONOZidmhOZHc1WGU4NHMzQ1AtMVE{#}gid=21

[54] Mike Bostock: D3.js - Data-Driven Documents (2019), https://d3js.org/

[55] MONICET: monicet :: Home (2009), http://www.monicet.net/en

[56] Nardelli, R., Dall, Z., Skevoulis, S.: Comparing tensorflow deep learning performance and experiences using CPUs via local PCs and cloud solutions. In: Lecture Notes in Networks and Systems, vol. 70, pp. 118–130. Springer (2020)

[57] Nether, T., Lange, G.: Laurens van der Maaten **1984**(August 2005) (2009), https://lvdmaaten.github.io/tsne/

[58] Newell, A.: A Textbook That Points the Way. Contemporary Psychology: A Journal of Reviews **26**(1), 50–51 (jan 1981). https://doi.org/10.1037/019640, http://access.portico.org/stable?au=phzkd9sbw

[59] Nocaj, A., Brandes, U.: Computing Voronoi Treemaps: Faster, Simpler, and Resolution-independent. Computer Graphics Forum **31**(3pt1), 855–864 (jun 2012). https://doi.org/10.1111/j.1467-8659.2012.03078.x, http://doi.wiley.com/10.1111/j.1467-8659.2012.03078.x

[60] OOM: OOM Website, https://oom.arditi.pt/

[61] OOM, Arditi: Cet PhotoID, http://oomdata.arditi.pt/prd/catalogofotoid/catalogo.php

[62] Parker, J.R.: Algorithms for image processing and computer vision. In: Algorithms for image processing and computer vision (1997)

[63] Project Jupyter: Project Jupyter | Home (2017), https://jupyter.org/http://jupyter.org/

[64] Redux: Getting Started with Redux · Redux (2019), https://redux.js.org/introduction/getting-started

[65] Sciences, N.: Finscan (2000)

[66] Shana Pearlman: What are APIs and how do APIs work? (2016), https://blogs.mulesoft.com/biz/tech-ramblings-biz/what-are-apis-how-do-apis-work/

[67] Sharma, S., Kaur, M.: Antimicrobial activities of rhizobacterial strains of Pseudomonas and Bacillus strains isolated from rhizosphere soil of carnation (Dianthus caryophyllus cv. Sunrise). Indian Journal of Microbiology **50**(2), 229–232 (jun 2010). https://doi.org/10.1007/s12088-010-0045-6, http://www.dtic.mil/docs/citations/ADA414520{%}0Ahttp://oai.dtic.mil/oai/oai?verb=getRecord{&}metadataPrefix=html{&}identifier=ADA414520{%}0Ahttp://www.dtic.mil/dtic/tr/fulltext/u2/a414520.pdf{%}0Ahttps://www.rand.org/pubs/documented{_}briefings/DB396.htmlht

[68] Stanley, R.: Darwin: Identifying dolphins from dorsal fin images. Senior Thesis, Eckerd College (1995)

[69] Supervisely: Supervisely, https://docs.supervise.ly/

[70] Tan, M., McDonald, K.: Bird Sounds by Manny Tan & Kyle McDonald | Experiments with Google (2017), https://experiments.withgoogle.com/bird-sounds

[71] Taylor Otwell: Laravel: The PHP framework for web artisans (2013), https://laravel.com/

[72] Temple, A.J., Kiszka, J.J., Stead, S.M., Wambiji, N., Brito, A., Poonian, C.N.S., Amir, O.A., Jiddawi, N., Fennessy, S.T., Pérez-Jorge, S., Berggren, P.: Marine megafauna interactions with small-scale fisheries in the southwestern Indian Ocean: a review of status

and challenges for research and management. Reviews in Fish Biology and Fisheries **28**(1), 89–115 (mar 2018). https://doi.org/10.1007/s11160-017-9494-x, http://link.springer.com/10.1007/s11160-017-9494-x

[73] The New Forest National Park: The New Forest National Park Location (2006), https://www.new-forest-national-park.com/new-forest-national-park-location.html

[74] Tosi, M.: [Clinical trial of bromazepam in internal medicine. Double-blind cross-over comparison with placebo]. La Clinica terapeutica **75**(6), 603–10 (dec 1975). https://doi.org/10.1109/CDC.2006.377042, http://www.icrwhale.org/pdf/SC0341-47.pdfhttp://ieeexplore.ieee.org/document/4177055/http://www.ncbi.nlm.nih.gov/pubmed/3309

[75] United Nations Development Programme: Goal 14 Life below water (2019), https://www.undp.org/content/undp/en/home/sustainable-development-goals/goal-14-life-below-water.htmlhttps://www.un.org/sustainabledevelopment/oceans/

[76] Vedaldi, A., Fulkerson, B.: Vlfeat. In: Proceedings of the international conference on Multimedia - MM '10. p. 1469. ACM Press, New York, New York, USA (2010). https://doi.org/10.1145/1873951.1874249, http://dl.acm.org/citation.cfm?doid=1873951.1874249

[77] Ventimiglia, T., Wayne, K.: The Barnes-Hut Algorithm, http://arborjs.org/docs/barnes-hut

[78] Wikipedia: Azores (2004), https://en.wikipedia.org/wiki/Azores

[79] Wikipedia: Bottlenose Dolphin (2004)

[80] Wikipedia: Cicada (2005), https://en.wikipedia.org/wiki/Cicada

[81] Wikipedia: Wikipedia (2005), https://en.wikipedia.org/wiki/Minke{_}whale

[82] Xue, Y., Wang, T., Skidmore, A.K.: Automatic Counting of Large Mammals from Very High Resolution Panchromatic Satellite Imagery. Remote Sensing **9**(9), 878 (aug 2017). https://doi.org/10.3390/rs9090878, http://www.mdpi.com/2072-4292/9/9/878

[83] Zanarini, P.: What is Information Overload? | Interaction Design Foundation (2018), https://www.interaction-design.org/literature/topics/information-visualizationhttps://www.interaction-design.org/literature/topics/information-overload

[84] Zelinsky, A.: Learning OpenCV—Computer Vision with the OpenCV Library (Bradski, G.R. et al.; 2008)[On the Shelf]. IEEE Robotics & Automation Magazine **16**(3), 100–

100 (sep 2009). https://doi.org/10.1109/MRA.2009.933612, http://ieeexplore.ieee.org/document/5233425/