FishEye

Marine Species’ Recognition and Visualization

Tiago Miguel Pedro do Nascimento

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisor: Prof. Sandra Pereira Gama

Examination Committee

Chairperson: Prof. José Carlos Alves Pereira Monteiro
Supervisor: Prof. Sandra Pereira Gama
Members of the Committee: Prof. Daniel Jorge Viegas Gonçalves

June 2017
Anyone who has never made a mistake has never tried anything new.

Albert Einstein
Acknowledgments

During the course of this dissertation as well as during my academic course, several people contributed in order to make it a more motivating journey.

First of all, I would like to thank my family and girlfriend for all their support, presence and motivation to give my very best.

I would also like to thank Sandra Gama, whose relentless guidance and support throughout this process have led to a higher quality work.

Finally, I would like to thank my friends and colleagues, for their support during the tests performed. A special thanks to Miguel Cruz for all its availability, support and critical spirit.
Abstract

Despite the historical limitations associated with the study of marine species, current technology makes it possible to collect ocean animal data in a more accessible way. Such data is crucial in a variety of contexts, both scientific and commercial. In fact, currently, with a variety of tagging and tracking devices, it is relatively easy to gather information on marine species. As a result, such information is nowadays present in large amounts, often in textual formats, making it difficult to interpret and analyze.

Information Visualization, due to its potential to represent large amounts of data while alleviating cognitive load associated with data interpretation, may overcome this limitation. Having this into consideration, we present a marine species’ visualization that allows the representation and interactive exploration of marine species’ telemetric data. It consists of an integrated dashboard that takes advantage of multiple visualization idioms which work together to provide the user with different, complementary views on the existing data. We performed usability tests which have pointed directions for our solution and validated its potential in making important patterns immediately perceivable, while allowing the user to take advantage of exploration and comparison mechanisms to obtain further relevant information.

Furthermore, a species’ image recognition module was implemented together with the described visualization, enabling users to proper recognize the species present on the picture followed by learning more about it, through the described visualization solution.

Keywords

Marine species’ recognition, Information Visualization, Integration, Species’ Comparison, Interactivity, Incremental and Iterative Development
Resumo

Apesar das limitações históricas associadas ao estudo de espécies marinhas, a tecnologia atual torna possível a recolha de dados de forma mais acessível. Este tipo de dados são cruciais em uma variedade de contextos, tanto científicos como comerciais. Atualmente, existe uma variedade de dispositivos de marcação e rastreamento, permitindo assim facilmente recolher informações sobre espécies marinhas. Como resultado, tais informações estão hoje presentes em grandes quantidades, muitas vezes em formatos textuais, dificultando a sua interpretação e análise.

A visualização de informação, devido ao seu potencial para representar grandes quantidades de dados aliviando ao mesmo tempo a carga cognitiva associada à interpretação dos mesmos, pode superar esta limitação. Tendo isto em consideração, apresentamos uma visualização de espécies marinhas que permite a representação e exploração interativa de dados telemétricos de espécies marinhas. É composto por um painel integrado que tira proveito de múltiplos idiomas de visualização que trabalham em conjunto para fornecer aos seus utilizadores diferentes visões complementares sobre os dados existentes. Testes de usabilidade foram realizados, sendo recolhidas orientações pontuais para melhorar a nossa solução e validando o seu potencial em fazer padrões importantes imediatamente perceptíveis, permitindo ao utilizador tirar proveito dos mecanismos de exploração e comparação existentes para obter o máximo de informação relevante.

Um módulo de reconhecimento através de imagens foi também implementado em conjunto com a visualização descrita, permitindo ao utilizadore reconhecer uma espécie e de seguida aprender mais sobre a mesma através do mecanismo de visualização descrito.

Palavras Chave

Reconhecimento de espécies marinhas, Visualização de Informação, Integração, Comparação de espécies, Interatividade, Desenvolvimento iterativo e incremental
## Contents

1 Introduction .................................................. 1  
  1.1 Objectives .................................................. 3  
  1.2 Document Structure .......................................... 3  

2 Related Work .................................................. 5  
  2.1 Recognition .................................................. 6  
    2.1.1 Feature Extraction ....................................... 6  
      2.1.1.A Scale-Invariant Feature Transform (SIFT) .............. 6  
      2.1.1.B Principal Component Analysis (PCA) .................... 7  
      2.1.1.C Gray Level Co-occurrence Matrices (GLCM) ............. 7  
      2.1.1.D Fourier Descriptors .................................. 7  
      2.1.1.E Gabor Filter (GF) .................................... 8  
      2.1.1.F Curvature Scale Space (CSS) .......................... 8  
      2.1.1.G GrabCut .............................................. 8  
      2.1.1.H Anchor/Landmark Points Location Detection .......... 9  
      2.1.1.I Gaussian Filter ...................................... 10  
      2.1.1.J Other Approaches .................................... 10  
      2.1.1.K Feature Extraction Discussion ......................... 11  
    2.1.2 Classification ........................................... 11  
      2.1.2.A Support Vector Machine (SVM) .......................... 11  
      2.1.2.B K-Nearest Neighbor (KNN) ............................. 11  
      2.1.2.C Neural Networks ..................................... 12  
      2.1.2.D Discriminant Analysis ................................ 12  
      2.1.2.E Zernike Moments ..................................... 13  
      2.1.2.F Mixed Approaches ................................... 13  
      2.1.2.G Classification Discussion ............................. 14  
  2.2 Visualization ............................................... 16  
    2.2.1 Geographical ............................................ 16  
    2.2.2 Temporal ................................................ 19  
    2.2.3 Mixed approaches ....................................... 21  
    2.2.4 Other approaches ....................................... 23
List of Figures

2.1 Moustahfid and Weise, [1], web portal graphical display that includes location data for track visualization in real-time. ............................................. 17
2.2 Moustahfid and Weise, [1], number of detection’s through time. ................................................................. 17
2.3 Halpin et al. [2], interactive fusion of oceanographic data (sea surface temperature) and models with animal observation data. ............................................ 18
2.4 Halpin et al. [2], leather-back turtle tracks off the coast of South Africa. ............................................................. 18
2.5 Zheng et al. [3], Global View for vulture migration shows the habitats of vultures in red. An OD-pair (i.e. Guatemala and Canada) is chosen for further exploration, indicated by blue and red markers. ............................................. 19
2.6 Zheng et al. [3], The OD-pair Flow View for taxi travel between Jing’an Temple and Lujiazui in Shanghai during a specific time. Three major bi-directional routes are labeled with I, II and III. ............................................. 20
2.7 Zheng et al. [3], Story-line View for vulture migration patterns. ........................................................................... 20
2.8 Silvestro et al. [4], scatter, statistical and density plots view. ............................................................................. 21
2.9 Ferreira et al. [5], system overview. .................................................................................................................. 22
2.10 Kitamoto et al. [6], the map view of capture records in 2008. The height of bars represents the number of bumblebees caught by participants at one place through the year. Users can zoom in or out to see the exact location, and the clicking of bars shows the list of records in an info window. ............................................. 22
2.11 Kitamoto et al. [6], a warning map created from survey data. Red color represents grids of high colonization probability, where green color represents the grid of low colonization probability. .............................................................................. 23
2.12 Kitamoto et al. [6], relationship between flowers and bumblebees. ............................................................................. 23

3.1 Clownfish (Amphiprion Percula) species. ........................................................................................................ 26
3.2 Powderblue Surgeonfish (Acanthurus leucosternon) species. .............................................................................. 26
3.3 Some Clownfish positive samples. .................................................................................................................. 27
3.4 Illustrative negative samples for recognition training. .......................................................................................... 28
3.5 FishEye’s species’ recognition’s module architecture. ......................................................................................... 29
3.6 FishEye’s recognized species’ welcoming screen’s modal regarding the Clownfish ........................................ 30
3.7 Powderblue Surgeonfish recognized within 250 positive samples. .............................................................................. 31
3.8 Powderblue Surgeonfish recognized within 250 positive samples. 

3.9 Clownfish correctly identified. 

3.10 Clownfish incorrectly identified. 

4.1 Clownfish initial data sample, taken from GBIF [7]. 

4.2 FishEye Data Sample 

4.3 FishEye physical characteristics data sample 

4.4 Low Fidelity Prototype - Radial Chart to represent time cyclical behavior. 

4.5 Low Fidelity Prototype - Mercator Projection Map and Depth evolution Line Chart 

4.6 Low Fidelity Prototype - Radial Bar Chart for countries distribution 

4.7 Low Fidelity Prototype - Radar Chart displaying information on species' physical characteristics 

4.8 Low Fidelity Prototype - species' physical characteristics location on a fish shape 

4.9 Low Fidelity Prototype - Bubble Word Cloud representing species’ keywords 

4.10 Functional Prototype - Spiral visualization representing time cyclical behavior 

4.11 Functional Prototype - Spiral visualization representing time cyclical behavior with further developments 

4.12 Functional Prototype - Mercator Projection Map showing species’ distribution 

4.13 Functional Prototype - Radial Bar Chart for countries distribution 

4.14 Functional Prototype - Treemap visualization displaying occurrences at continent level and for Oceania respectively 

4.15 Functional Prototype - Radar Chart visualization displaying fish species physical traits information 

4.16 Functional Prototype - Treemap and Projection Map linking and interaction mechanisms example 

4.17 Functional Prototype - BeeSwarm implementation 

4.18 Functional Prototype - FishEye’s first Interface layout 

4.19 Functional Prototype - FishEye’s Interface layout improvements 

4.20 Functional Prototype - FishEye’s further development interface improvements and spiral’s detail mode 

4.21 FishEye’s final layout 

4.22 FishEye’s species’s comparison - species list 

4.23 FishEye’s species’s comparison mode 

4.24 FishEye’s species’s comparison mode regarding Spiral’s detail view 

4.25 FishEye’s architecture schema 

4.26 FishEye’s dashboard regarding Surgeonfish species' 

5.1 Box Plot for Question solving duration (s) 

5.2 Grade rankings of SUS scores based on the works of Miller et al. [8] 

5.3 User answer distribution for the System Usability Scale (SUS) questionnaire.
## List of Tables

<table>
<thead>
<tr>
<th>Section</th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Studied approaches results</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Visualization studied approaches overall characteristics</td>
<td>24</td>
</tr>
<tr>
<td>3.1</td>
<td>Recognition metrics comparison between first and final development iterations</td>
<td>30</td>
</tr>
<tr>
<td>3.2</td>
<td>FishEye evaluation results</td>
<td>33</td>
</tr>
<tr>
<td>4.1</td>
<td>Violated heuristics frequency</td>
<td>51</td>
</tr>
<tr>
<td>4.2</td>
<td>Violated heuristics severity frequency</td>
<td>51</td>
</tr>
<tr>
<td>5.1</td>
<td>Users tests duration for each question</td>
<td>63</td>
</tr>
<tr>
<td>5.2</td>
<td>Users number of errors made for each question</td>
<td>66</td>
</tr>
<tr>
<td>5.3</td>
<td>Users Usability Scale (SUS) item rating</td>
<td>67</td>
</tr>
<tr>
<td>5.4</td>
<td>Wilcoxon signed rank test results for response time metric</td>
<td>68</td>
</tr>
<tr>
<td>5.5</td>
<td>Pearson’s correlation coefficients between response time and errors</td>
<td>69</td>
</tr>
</tbody>
</table>
Introduction

Contents

1.1 Objectives .................................................... 3
1.2 Document Structure ....................................... 3
Human limitations associated with the study of underwater ecosystem’s have made it difficult to study marine species. However, with the continuous growth of technology, marine animal data collecting has become more accessible, presenting growing potential for knowledge acquirement regarding these species. Such knowledge is important in a variety of contexts, both for scientific and commercial purposes.

Currently, significant investment has been made in tagging and tracking technology. Resulting data, if properly analyzed, may help to understand intrinsic and external factors that may influence marine species’ behavior, leading to a better overall understanding of the ecosystems in which such animals are inserted. Furthermore, adequate analysis of this information may reveal a number of patterns relative to different species, aiding in the process of understanding the behavior of these animals. Nevertheless, telemetry data gathered through animal tagging presents several challenges that make it hard to analyze. Not only does it often consist of large amounts of information, but it is also frequently found in a textual form, difficult to read and understand. As a consequence, deriving knowledge from this information is often a long and extremely demanding process. For instance: (1) Does Oceania in fact have a larger number of recorded clownfish occurrences when compared to the rest of the world? (2) Which species is more often recorded in colder months? (3) Has the surgeonfish inhabited deeper ocean areas over time?

One way to overcome this challenge is through the use of Information Visualization (InfoVis), which alleviates cognitive load associated with data interpretation [9]. A meaningful visualization will in fact allow the representation of information in a way that highlights relevant information while providing the means for the user to explore and find important patterns.

Having the described limitations into consideration, FishEye was created, a visualization which presents marine species’ information in an integrated manner while providing users with new means of learning about fish species and at the same time interacting with such data. Furthermore, it allows the comparison of two species simultaneously across all existing information domains, enhancing the identification of common behaviors, as also, establishing relations about such species.

FishEye consists of a dashboard with multiple, interconnected views that provide complementary perspectives on the presented information. These views change dynamically according to user input, highlighting relevant contextual information and informing the remaining idioms of changes that are taking place.

An iterative and incremental paradigm was followed during development, hence performing an expert evaluation with a set of heuristics suited for information visualization [10] which, while pointing directions for the ongoing development of our solution, rendered it valid for further development and subsequent testing. Furthermore, to understand how, in a real life context, users react and experience FishEye, quantitative usability metrics were gathered by performing usability tests.

In order to possibilitate users to identify fish species to further learn about them, a species’ recognition module was integrated within FishEye. This consists of extracting features about the provided species and then classifying, successfully recognizing it. To proper validate the recognition performance, an evaluation to the described module was carried out which showed FishEye’s potential to
1.1 Objectives

The main goal of this dissertation, is to present marine species' information in an integrated manner while providing users with new means of learning about fish species and at the same time interacting with such data.

To achieve such goal a set of requirements must be fulfilled, as follows:

The information must be contained on a single page and accessible among the several existing devices and operating systems. Such information must also be relevant and meaningful allowing users identifying patterns, inferring knowledge and give a better understanding on species behavior as also being able to compare multiple species.

Species' data must be displayed in multiple views and address multiple domains, while maintaining context across all views and linking them together. The default displayed information must follow Shneiderman's mantra for visual data analysis [11]: overview first, zoom and filter, then details-on-demand. Interaction mechanisms must also be available allowing the user to further explore and detect patterns in a more efficient manner, simultaneously engaging and potentiating the user a more enjoyable experience [9]. A personal and customizable experience must be given to users when interacting within the information allowing to manipulate the existing layout and information as they please.

The most modern approaches on both species’ recognition and information visualization were studied and discussed, identifying existent limitations, which serving as motivation and foundation for the proposed solution.

1.2 Document Structure

This document is organized as follows. In chapter 2 we introduce relevant related work that situates our study on both species recognition and information visualization most recent breakthroughs. We then present and detail our solution's both recognition process, in chapter 3, and visualization in chapter 4, which is followed by a description of the performed user usability evaluation in order to validate the work done. We then draw the main conclusions regarding all the work done and a set of guidelines for further improvements on the current status of our marine species' visualization.
As approached in the previous chapter, this research has two main areas of interest that complement each other, fish species recognition and their respective information visualization, merging them together in a reliable solution that not only has the capability to recognize species but also the power to teach us about it. In the following subsections the recent breakthroughs in these two fields will be described and discussed, giving insights on what have been already accomplished and how it can be improved.

2.1 Recognition

The recognition of fish species can be divided in two major phases, feature extraction in which fish species unique characteristics that differentiates them among each other are collected, and fish classification that uses the gathered features and compares them with a knowledge base of previously gathered features trying to find a match, and so being able to efficiently discover that particular fish species. In the following subsections more detailed information is given about these two phases, the most popular algorithms and approaches and how they have been used in recent years.

2.1.1 Feature Extraction

Feature extraction is the method of capturing the most relevant characteristics from a given input. Its main goal is to extract the data that is non-redundant and informative, creating a less complex but equally comprehensive representation of the data to work with. In the context of this study, it represents the process of first, extracting a fish from a given image (segmentation), and then extract features that represent the fish itself, enhancing classification performance and accuracy. The features extracted in recent similar works can be divided in three major groups: Color, Geometric/Shape and Texture features. Despite sharing the same goal, different approaches have been followed by the authors of these solutions, being the main ones: Scale-Invariant Feature Transform (SIFT), Principal Component Analysis (PCA), Gray Level Co-occurrence Matrices (GLCM), Fourier Descriptors, Gabor Filter (GF), Curvature Scale Space (CSS), Connected Components algorithm, GrabCut, Anchor/Landmark points location detection, Histograms, and Gaussian filters described with more detail in the following subsections. After their individual description and analysis a comparative table between them all is provided serving as base for discussion.

2.1.1.A Scale-Invariant Feature Transform (SIFT)

SIFT is a reliable feature detector that is used to match between the possible different views (invariant scale, orientation, illumination and noise) of the same object. The technique consists of the following four steps defined by Lowe [12]: scale-space extrema detection, keypoint localization, orientation assignment, and keypoint descriptor [13–16]. However SIFT presents some limitations, it uses a one-dimensional (1D) vector of scalar values as a local feature descriptor and cannot be extended to operate on color images which generally consist of three-dimensional (3D) vector values, the processing of the algorithm is computationally expensive.
causing it to be slow and not the best choice for real-time applications. Also, SIFT algorithm was strongly affected by the 3D rotation of individuals in Rodrigues et al. [16] experiments, dropping the accuracy drastically.

2.1.1.B Principal Component Analysis (PCA)

PCA is a standard technique for dimensionality reduction. It is a simple, non-parametric method of extracting relevant information from confusing data sets reducing its complexity.

It was demonstrated that PCA was well-suited to parameterize shape, appearance and motion of fish species [16]. The goal of using this feature extracting technique was to extract the principal components of higher variance from images, more specifically the color information encoding, and evaluate their impact in the overall accuracy of Rodrigues et al. [16] framework.

Lopez-Villa et al. [14] have used a visual bag-of-words model to quantize each descriptor vector into one visual word, and represent each image by a histogram of visual words [17]. The PCA was used to reduce this model size, retaining 90% of the input variance. This reduction method was also applied in the work by Spampinato et al. [18], employed on a 120 extracted features vector, PCA allowed them to select only the relevant 24 features for further classification stage. Although PCA is quite simple, it presents some shortcomings, such as its implicit assumption of Gaussian distributions, its restriction to orthogonal linear combinations and not being scale invariant [16].

2.1.1.C Gray Level Co-occurrence Matrices (GLCM)

Used for texture features, the GLCM is a statistical method which considers spatial relationships of gray level image pixels. It describes the co-occurrence frequency of two gray scale pixels at a given distance. Although applied in different works such as Khotimah et al. [19], Huang et al. [20], Hu et al. [21] and Spampinato et al. [18], their goal is the same when applying this technique, to extract information about several statistical features like Energy, Correlation, Inertia, Entropy, etc. However, GLCM efficiency reduces in a significant way when there are a large number of gray levels on the given image, since it cannot properly handle color composition and color spatial information.

2.1.1.D Fourier Descriptors

Fourier descriptors are a way of encoding the shape of a two-dimensional object by taking the Fourier transform of its boundary. The original shape can be recovered from the inverse Fourier transform. However, if only a few terms of the inverse are used, the boundary becomes simplified, providing a way to smooth or filter the boundary[4]. They are independent of the object’s position, orientation, scale and slant.

This method was used to extract information about the bounding contour of the fish head and body [13, 18, 20, 22].

To be efficient, Fourier descriptors must be applied on an already segmented image and a high number of descriptors should be used.

2.1.1.E Gabor Filter (GF)

Gabor is a edge detection filter that relies on frequency and orientation, Shrivakshan and Chandrasekar [23] claim these are similar to those of the human visual system. This filter was used in order to get texture information from the fish image. Its was applied so that new quality features like Standard Deviation and Mean could be obtained [18, 20, 24].

2.1.1.F Curvature Scale Space (CSS)

CSS is a shape feature extractor, similar to the Fourier descriptors method, its goal is to get the contour of an object. It had different applications: to calculate the fish orientation by weighting each contour pixel with its local curvature scale, and then to align all fish horizontally where the head of the fish is located on the right. This is based on a streamline assumption, which assumes that most fish have a smoother head than tail [20], utilized to extract the first 20 local maxima (that are curve length values) of the CSS image, normalized by the fish contour length [18], and also was used to detected non-fish objects on the photos and reject them [25].

However such technique has some drawbacks defined by Sai et al. [26], such as poor performance with deep and shallow concavities of the shape and failing to address the problem of open curves present in the given shape.

subsubsection Connected Components Algorithm This algorithm is used to detect connected regions in digital binary images. Given a heuristic, subsets of connected components can be uniquely labeled enabling blobs to be extracted and/or detected from the resulting binary image Fabic et al. [27]. After being captured the fish outline is converted to a blob using this method [27]. Other usages include attribute size estimation [22] and identifying each segmented object [25, 28].

Some problems may arise when two different regions overlap each other getting mislabeled as one unique region.

2.1.1.G GrabCut

GrabCut is used as an image segmentation technique based on graph cuts, this process can be done automatically or with user interaction if needed [29]. It’s with this purpose of segmentation that this algorithm was used on the studies by Chuang et al. [13] and Huang et al. [20].

Despite using object segmentation masks that were provided along with the dataset used, Chuang et al. [13] claim that when needed this technique can be used to generate segmentation masks automatically in real-time.

Some failures can happen when using this method, the accuracy of segmentation drops when low contrast images at the transition from foreground to background are used and when the true foreground and background distributions overlap partially in color space [29].
2.1.1.H Anchor/Landmark Points Location Detection

Point detection is applied to find a significant set of points that will help in obtaining the anchor measurements for patterns of interest [24]. The goal of applying such method was to determine a number of labeled points (23 and 17 respectively) that will give the location of each feature determined for fish recognition [24, 30]. After, its used to calculate new features such as size and shape measurement features.

Some limitations were found on the usage of this technique, Alsmadi et al. [30] affirms that 14 from the 17 anchor points are established manually and only the remaining three automatically, also, this approach received little attention due to the fact that it required an additional step of reliably locating fish landmarks/anchor points, which may affect their overall performance. Usama and Alsmadi [24] gives no detail about the automation of locating such points.

Histograms

A histogram is a graphical representation of the distribution of numerical data. It estimates the probability distribution of a given variable by dividing the entire range of values into a series of intervals and then count how many values fall into each interval. In the SIFT algorithm, orientation histograms are computed from the gradient orientations of sample points within a region for each interest point prior identified. This histograms contain eight bins each, and each keypoint contains a 4x4 array of 16 histograms around it. The purpose of this step in SIFT is to enhance invariance to changes in illumination [13–16].

Other works used histograms to analyze color features, grayscale histograms (GH) are used to store statistical information about the brightness attribute in a given image. New features including mean intensity, mean contrast, roughness, etc, are then extracted based on this histogram [18, 21].

In the research by Huang et al. [20], normalized color histograms in the Red and Green channels and Hue histogram of the HSV color space are used to minimize the effect of illumination changes.

Fabric et al. [27] also used this method, but for for segmentation purposes, sample Blue and Non-Blue histogram templates were gathered over several frames of video sequences, these were then used to generate the respective mean values for the two types of image templates. Then for every block in the image frame to be processed, color histograms are obtained and compared with the previous mean values calculated. The Non-Blue blocks are blackened while the Blue blocks are left unchanged. However this only works for that particular oceanic area where the video samples were recorded.

Other color models were also tried, a HMMD color model was developed from the RGB and the HSV color spaces. Then a 128 bins color histogram was used as the object feature for the classification stage [28].

A shape histogram was also used featuring Fourier descriptors [18].
2.1.1.1 Gaussian Filter

In the context of image processing, the Gaussian filter is used to reduce noise and detail from an image by convolving it with a kernel of Gaussian values. It is used as a sub-step of the SIFT algorithm\[14, 16\] where the image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images are taken. It is also used as an initial step of the Canny Edge Detector\[27\] and it is applied on shape contours reducing noise and smoothing it\[13, 18, 20, 22, 25\].

2.1.1.2 Other Approaches

Not as common as the above subsections, but equally important, other approaches have also been followed regarding feature extraction.

Some techniques to help the process of segmentation were used, Inward-Outer Block Erasure algorithm, a block-level scanning algorithm from the left to right and downwards starting from the topmost left block. In which targeted blocks are histogrammed and compared to their outer neighbors. If their outer neighbors have mean histogram values that are black, they are also blackened. Edge Cleaning algorithm, a method that performs a pixel-wise scan that checks if the previous algorithm missed any region. Finally, Canny Edge detector is used for detecting fish contours\[27\].

A wavelet transform for texture features was also tested and obtained better results than the former used statistical method\[21\].

For means of comparison with the SIFT approach, Speeded Up Robust Features (SURF) algorithm was used\[15\], obtaining better results then SIFT, SURF complexity is low since it fixes the repetitive orientation using information from a circular region around the interest point. Then, the algorithm constructs a square region aligned to the selected orientation, and extract the descriptor from it\[31\].

For the background subtraction from video frames\[14\] the SubSense algorithm that consists of a universal pixel-level segmentation method that relies on spatio-temporal binary features as well as color information to detect changes was used\[32\].

Also on this topic, to extract the fish image object the difference between the current video frame and a background reference model was calculated\[28\], however this is very limiting since it only works for this particular reference model.

An adaptive thresholding technique using Otsu’s method\[33\] was employed to generate an initial binary mask of foreground objects, then watershed segmentation with an automatic marker generation scheme\[34\] is used to extract the boundary around each fish. To finalize, a refined binary segmentation result is generated by subtracting the watershed segmentation from the initial mask. In order to remove the factor of illumination, pixel values within object region are also taken into account. The average of pixel values within an object is calculated and converted to normalized-RGB color space\[25\].
2.1.1. Feature Extraction Discussion

Feature Extraction is an essential step before moving for the classification phase. Many approaches, some more popular than others, were used along the referenced works, however with the exception of the work by Khotimah et al. [19] where the segmentation stage had an accuracy of about 96.9%, no more authors give information about accuracy and time spent when applying such techniques, making them difficult to compare.

2.1.2. Classification

Classification is, in the context of this study, the process of differentiate and recognize fish species according to some shared feature characteristics among its species [16]. Several algorithms have been proposed and applied in recent works. In the following subsections an overview about these techniques such: Support Vector Machine, K-Nearest Neighbor, Neural Networks, Discriminant Analysis, Zerknike Moments and mixed approaches will be described. After, a comparison table (Table 1) between them is presented and discussed. All presented works follow the same process for testing its classification power, using an available dataset of fish images/videos, first the classifier is trained and only after is tested. All this information is available in the results table (Table 1) mentioned before.

2.1.2.A. Support Vector Machine (SVM)

SVM’s are used for recognizing patterns or discriminating between two groups [35]. It solves the classification problem by trying to find an optimal separating hyperplane between classes [15]. However the need for classifying more than two species, asks for a Multi Support vector Machine (MSVM), to enable this multi-class classification, several mechanisms, such as one-against-one (OAO) and one-against-all (OAA), have been developed. This classification is done simultaneously around all classes and the possible inter-class correlations are omitted [15].

Two types of OAO based MSVM’s were constructed and compared in the work by Hu et al. [21]: directed acyclic graph MSVM (DAGMSVM) and voting based MSVM (VBMSVM), these were compared with the benchmark Library for Support Vector Machines (LIBSVM). After experimental testing DAGMSVM was the method which got better results when combining accuracy and processing speed. Other works also used SVM’s although they were used in conjunction with other techniques. Such works are described in Section 2.1.2.6.

2.1.2.B. K-Nearest Neighbor (KNN)

Nearest neighbor is a popular non-parametric method used for classification of objects based on attributes and training samples. The NN consists of a supervised learning algorithm where the result of a new instance query is classified based on the majority of the nearest neighbor category. Nearest neighbor is not negatively affected when training data is large and noisy [15].

In the research by Rodrigues et al. [16], before applying this classification algorithm, Euclidean distances between the feature vectors of the input image and the vectors on the knowledge base
created during the training phase are computed. Then the two features extraction techniques applied, PCA and SIFT, are compared to conclude which one has better results. For the PCA a simple 1-NN was used, meaning that the species of an individual is determined as being the one associated to the smallest Euclidean distance found. Regarding SIFT method, a 2-NN was used following the premise that a correct match implied having the closest neighbor significantly closer then the closest incorrect match.

A 3-NN classifier is used in the work by Lopez-Villa et al. [14], the authors only express they have used it on an histogram of visual words for their classification stage and no more information is given regarding the usage of such method.

Similarly to these two works, Leel et al. [28] applied this method on the HMMD color space histograms computed on the feature extraction phase. On experimental testing, 1-NN, 3-NN and 5-NN were used and compared, being the 1-NN method the one that achieved better results.

Some limitations of this algorithm are the need to determine parameter K (number of nearest neighbors), calculate the distances between the query instance and all the training samples, sort this distances and determine the nearest neighbors based on the \( K^{th} \) minimum distance, as well as determining the categories of the nearest neighbors [15].

### 2.1.2.C Neural Networks

Neural networks are non-linear statistical data generalized mathematical models that exhibit biological nervous systems, which inspires the learning process of the human brain [36], mainly used in problems where it is hard to find correlation between the input parameter and the outputs. The functions of a biological neuron are modeled by computing a differentiated nonlinear function. It aggregates different inputs from other neurons or the external environment to form an output [37].

A multilayer feed forward neural network model with back propagation algorithm for training is employed for classification containing three layers with a varied number of neurons, the input, hidden and output layer [30].

A different type of Neural Network is used by Li et al. [38], a Fast Regions with Convolutional Neural and Networks (Fast R-CNN) is proposed for classification of fish species. The main idea behind this neural network is to take as input a RGB image and its 2000 regions of interest collected by selective search and produces a distribution over fish classes as well as related bounding-box values.

To be efficient, Neural Networks need to be trained within a vast domain and this domain should not be explained by a limited number of features [39].

### 2.1.2.D Discriminant Analysis

The Discrimant Analysis consists of a sequence of methodologies that, given a k-dimensional set partitioned into a number of classes tries to find a linear combination of features that characterizes or separates two or more classes [18].
Applied in the work by Spampinato et al. [18], the used dataset was divided in five subsets, four for training and the remaining one for testing and then the algorithm is applied for each subset.

Complications arise when the number of features for each class surpass the total number of classes used in that respective subset [18].

2.1.2.E Zernike Moments

Image moments are a powerful shape descriptor that captures the global features of objects. These are widely used in the field of image pattern recognition due to their invariant properties. The set of orthogonal Zernike moments are known to be superior compared to other image moments due to their rotational, translational and scale invariant properties [40].

Applied in the research by Fabic et al. [27], in the training phase templates from known species of fish are gathered. The Zernike moments up to $12^{th}$ order are calculated for each template. These moments are from different combinations of order and repetition. Then, averages of the moments with the same order and repetition of each species template are calculated and each species average is subtracted from another to get the average difference distance. This result is then halved to get the Zernike moment threshold for each species.

In the testing phase, the moment of the tested fish is subtracted from the ones computed in training, then the top five smallest differences are compared with the computed Zernike moment threshold. If none of them exceed the threshold, the fish specie is determined by the majority of type of fish correlated with the differences prior calculated. Otherwise, the fish can not be identified.

2.1.2.F Mixed Approaches

Some authors investigated more than one technique at a time in order to guarantee the best results.

In the work by Fouad et al. [15] three machine learning classifiers, namely Artificial Neural Network (ANN), Support Vector Machines (SVMs), and K-Nearest Neighbor (K-NN), are applied to the classification stage. Despite no detail is given on how these three techniques were applied, the best accuracy, while testing, is achieved when the SVM classifier using Gaussian radial basis function (RBF) as kernel function is applied.

A class hierarchy that follows a binary tree structure is purposed in the researches by Chuang et al. [13, 22]. First the data is all separated within two clusters using the expectation-maximization (EM) algorithm for Mixture of Gaussians (MoG). Then for each species, data is relabeled based on which cluster the majority of this species belongs to. A SVM with RBF kernel function is trained with these two clusters. The above steps are then repeated separately within each cluster until there is only one species by cluster.

A generic fish classification using a hybrid meta-heuristic algorithms (genetic algorithm with iterated local search) with back-propagation algorithm (GAILS-BPC), to classify the images of fish into dangerous or non-dangerous families, is proposed. A Genetic Algorithm (GA) is a population based
heuristic approach that simulates the procedure of natural selection and iterated local search is a search procedure that instead of searching in whole search space of solutions, it searches within a local subspace, this is the local where optimal solutions lies [24].

In another approach, two MSVM are created for binary classification problem, using one-vs-one strategy with a voting mechanism. The first classifies all classes simultaneously and will serve as baseline for comparing with the second and proposed technique, where a tree is constructed that organizes all the classes hierarchically thus helping resolve the error accumulation issue and making use of the inner-class similarities among fish species. In experimental testing, the second method proved itself faster and more accurate then the baseline technique [20].

2.1.2.G Classification Discussion

A variety of methods have been used to tackle the classification of fish species, being the most popular SVM, and K-NN between the referenced works. In Table 1, an overall view of the conditions in which these work were tested and also the achieved results is presented.

<table>
<thead>
<tr>
<th>Studied approaches</th>
<th>$N^o$ of identified species</th>
<th>Training samples</th>
<th>Testing samples</th>
<th>Total samples</th>
<th>Time (ms)</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rodrigues et al. [16]</td>
<td>4</td>
<td>24</td>
<td>24</td>
<td>48</td>
<td>-</td>
<td>92%</td>
</tr>
<tr>
<td>Fabic et al. [27]</td>
<td>2</td>
<td>-</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>50%</td>
</tr>
<tr>
<td>Hu et al. [21]</td>
<td>6</td>
<td>270</td>
<td>270</td>
<td>540</td>
<td>12.7</td>
<td>97.77%</td>
</tr>
<tr>
<td>Chuang et al. [22]</td>
<td>7</td>
<td>-</td>
<td>-</td>
<td>1325</td>
<td>-</td>
<td>90%</td>
</tr>
<tr>
<td>Huang et al. [20]</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>3179</td>
<td>several hours</td>
<td>90%</td>
</tr>
<tr>
<td>Usama and Alsmadi [24]</td>
<td>N.A</td>
<td>220</td>
<td>100</td>
<td>320</td>
<td>-</td>
<td>80.5%</td>
</tr>
<tr>
<td>Spampinato et al. [18]</td>
<td>10</td>
<td>220</td>
<td>100</td>
<td>320</td>
<td>-</td>
<td>90%</td>
</tr>
<tr>
<td>Fouad et al. [15]</td>
<td>1</td>
<td>91</td>
<td>60</td>
<td>151</td>
<td>-</td>
<td>100%</td>
</tr>
<tr>
<td>Alsmadi et al. [30]</td>
<td>N.A</td>
<td>257</td>
<td>93</td>
<td>350</td>
<td>-</td>
<td>86%</td>
</tr>
<tr>
<td>Khotimah et al. [19]</td>
<td>3</td>
<td>48</td>
<td>12</td>
<td>60</td>
<td>-</td>
<td>88%</td>
</tr>
<tr>
<td>Chuang et al. [13]</td>
<td>15</td>
<td>-</td>
<td>-</td>
<td>26.418</td>
<td>-</td>
<td>97.7%</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>-</td>
<td>-</td>
<td>2.195</td>
<td>-</td>
<td>98.4%</td>
</tr>
<tr>
<td>Lopez-Villa et al. [14]</td>
<td>5</td>
<td>-</td>
<td>-</td>
<td>1.573</td>
<td>-</td>
<td>93.45%</td>
</tr>
<tr>
<td>Leel et al. [28]</td>
<td>6</td>
<td>480</td>
<td>120</td>
<td>600</td>
<td>-</td>
<td>90%</td>
</tr>
<tr>
<td>Chuang et al. [25]</td>
<td>N.A</td>
<td>-</td>
<td>-</td>
<td>159</td>
<td>-</td>
<td>94.5%</td>
</tr>
<tr>
<td>Li et al. [38]</td>
<td>12</td>
<td>8.233</td>
<td>7.817</td>
<td>16.050</td>
<td>311</td>
<td>81.4%</td>
</tr>
</tbody>
</table>

Despite, the majority of the referenced works having accuracy rates greater or equal to 90%, time information is only given in two of them, in the work by Huang et al. [20] this component has a very time-consuming result, showing it won’t be worth using its techniques in further stages of this study since its to create a real-time solution, however in the research by Hu et al. [21] such opportunity is at reach within a fast time result. In the work by Fouad et al. [15] an 100% accuracy is achieved by this system, although since it was only applied for the classification of one fish species is hard to make conclusions about its scalability for classifying new species, although is worth applying it to the

Classification isn’t done regarding species but instead the type of fish, i.e, poisonous fish / non-poisonous fish
purpose of this study. One other problem that is present in the approaches where more than five fish species were classified is the fact that the dataset of fish samples used is highly unbalanced between classes, meaning that some species have much more samples than others.
2.2 Visualization

With the growth of technology, it has become easier to collect animal data. This data can be used to explore the intrinsic and external factors that may influence animal behavior [41], and so obtain a better understanding of the ecosystem function and evolutionary constraints of species [1]. However, the steps needed to go from data collection to being able to draw insights and conclusions about the data itself, such as organization, exploration, visualization, quantification, inference and generalizations, can be a long and extremely demanding process [41].

On the following subsections some recent approaches and techniques applied on this subject will be discussed, being the main topics about geographical and temporal distribution of species. After a table (Table 2) that focus on all works main characteristics for comparing purposes will be discussed.

2.2.1 Geographical

The study of movement has gained great momentum in recent years, and also more species telemetry data is being collected from year to year through sensors. This particular type of data helps to understand the distribution and space use of species, migration patterns, spread of diseases, among others, [41].

In the research by Moustahfid et al. [1], a web interface system provides graphical displays of ocean profile data and location data for track visualization in real-time. The visualization is done on top of a Google Maps based user interface, giving the user the capability to interact with the data such as panning and zooming. The default view of the interface shows the most recent data (users can select from 10, 60, or 90 day displays), and the interface features a pull out data menu which allows users to view or hide datasets from each species by using check boxes, (Fig. 2.1). By clicking on an animal icon, the user can also obtain additional information by means of a tooltip.

Nevertheless, this interface presents some limitations: there is no explanation about the meaning of the color coding used, new visualizations are done on top of the previous ones, sometimes causing overlapping and lost of context (Fig. 2.2), the tooltips are unnecessarily large and flat causing occlusion of the elements behind them, new Visualizations pop-up on top of the last Visualization causing occlusion (Fig. 2.2), no information about the system’s performance is given, such as the time it takes to render the visualizations and how its behavior according to the user’s checking/unchecking of the check boxes.

A similar approach is used by Halping et al. [2], using telemetry data to analyze and display information about movement patterns and distribution of sea animals is done using Google Maps interface. Analysis of marine animal distributions often requires analysis of the underlying oceanographic conditions at the time of the observations. This system permits such mechanism, having a direct linkage to common ocean data layers, allowing further deep analysis (Fig. 2.3). In this system any person can provide data increasing the risk of misleading visualizations. Similar to Fig. 2.3, where bubbles are overlapping in some cases, on the movement patterns’ visualization window (Fig. 2.4), cluttering issues grow proportionally to the number of species visualized, showing the system has scaling prob-
Figure 2.1: Moustahfid and Weise, [1], web portal graphical display that includes location data for track visualization in real-time.

Figure 2.2: Moustahfid and Weise, [1], number of detection’s through time.

lems. Having an abundant number of data attributes, derived measures could provide new species relevant insights, however the authors don’t seem to take advantage of such method.

The OzTrack project [42], also provides integration and overlay of the species tracking data with related environmental data. Beyond similar interactions described previously, users can also display the raw values and statistical values (derived measures) of the attributes on a panel near the visualization window and they can choose between a diverse number of different techniques to display the data from which he can choose. The system was tested intensively by a diverse range of people.
Figure 2.3: Halpin et al. [2], interactive fusion of oceanographic data (sea surface temperature) and models with animal observation data.

Figure 2.4: Halpin et al. [2], leather-back turtle tracks off the coast of South Africa.

from students to government officials where limitations have been pointed out such as: the level of available metadata being very limited, making difficult to take deep insights of the animal behavior, animals trajectories tending to overlap and the lack of error/uncertainty data information. One other major drawback of this project is the fact that the visualizations take a considerable time to be ready, being the users provided with a URL that they can keep checking to know if the rendering is ready to
be shown.

A visual analytics system is proposed in the work by Zheng et al. [3], allowing users to interactively analyze bi-directional movement behaviors with different levels of detail between three different linked visualization windows, a global View, a flow View and a story-line View, is provided. The following scheme, overview first, zoom and filter, then details on demand, defined by Shneiderman [43] was followed. Users first start with the Global View (Fig. 2.5), which provides an overview of Origin-Destination (OD) pairs with diverse bi-directional movement behaviors. Users are free to explore any region on the map and choose two regions of their interest as an OD-pair for further exploration. Then, the Flow View (Fig. 2.6) is provided to present all routes in two directions for the chosen OD-pair and to investigate the distribution of the spatial and mobility-related factors. Furthermore, if users are interested in a certain route, the Story-line View (Fig. 2.7) will visualize detailed information (e.g. time spent, speed) of individual’s behaviors along the route and allow users to explore and compare the micro patterns in two directions within a certain time period.

![Figure 2.5: Zheng et al. [3]. Global View for vulture migration shows the habitats of vultures in red. An OD-pair (i.e. Guatemala and Canada) is chosen for further exploration, indicated by blue and red markers.](image)

### 2.2.2 Temporal

Although the time attribute is present in the previous section, its role is only to filter data for the geographical representations.

However, in [4] it has a greater purpose. To address multivariate and time-dependent data, Silvestro et al. [4] designed a system that supports several, complementary views in which the user can choose from a list of four different views, being the first three represented on a single view on
Figure 2.6: Zheng et al. [3], The OD-pair Flow View for taxi travel between Jing’an Temple and Lujiazui in Shanghai during a specific time. Three major bi-directional routes are labeled with I, II and III.

Figure 2.7: Zheng et al. [3], Story-line View for vulture migration patterns.

(Fig. 2.8): (i) Scatter plots, where any data field can be used as x-axis and y-axis. An extra third data field can also be used as a category to accordingly color each data point in the scatter plot. When data overlapping occurs making the scatter plot difficult to understand, transparency can be used to reduce visual clutter. Moving the mouse pointer over the plot, details on data points are displayed next to the view. (ii) Statistic Metrics, a line chart of mean values as well as standard deviation can be superimposed on the scatter plot diagram, showing aggregated statistical quantities. (iii) Density plots, the density of data samples in the dataset for each y-axis value is calculated and shown as a histogram. (iv) Multiple Timelines View, different line charts are drawn on the same plot, representing the mean value curve for aggregated data on different time intervals.

Some issues were found on this work, such as the scalability and reading problems on the mixed plots view such as problems with visual cluttering and also some color coding information is missing in some views.
2.2.3 Mixed approaches

In the work by Ferrerira et al. [5], a system for visualizing bird distribution models across space, time and species is provided (Fig. 2.9). The system is organized in a number of different views that is decided by the user, being each view independent of each other, making it easy to compare information between different species. For geographical features to visualize migration patterns, including timing, direction, speed, and duration of movements the system provides an occurrences and variations map, in which the user is given the option of choosing the color gradient. For temporal features the user is given in a simple line plot the number of occurrences of the chosen species across an established time frame. Each possible habitat is defined by 16 characteristics, being each used as a tag in a tag cloud and its importance for that specific time-space is given by font size. The user can interactively drag the tag cloud on the map, and its content will be updated accordingly. Beyond some possible user interactions already described, the user can also interact with the views by panning, zooming and change the time-frame with the keyboard, affecting all views simultaneously. However some limitations rise, the length of the tags can influence their perceived sizes, becoming hard to say how important is a tag over another, to change the dates (the user has to go through all dates to reach a goal date, while an input option like a calender would serve better), the variation map is static for that specific time-frame, such type of information should be dynamic such as timeline animation. Giving the user the choice to choose the color coding may be an advantage and a disadvantage at the same time if the user chooses poorly.

On a different approach, a monitoring system for evasive species [6], in particular the *Bombus terrestris* bumblebee, because of their potentially wide spatial and temporal coverage is proposed. The website contains different types of views, some as depicted on Figs. 10, 11 and 12, nevertheless they are not linked in any way and locating them through the site menu’s is hard. These views are quite simple, being the main ones: (i) A map view (Fig. 2.10), defined for either a set of records
or an individual record to illustrate the geographical distribution of the capture records. This view can also be used as warning map (Fig. 2.11) to compare the number of bumblebees caught and the risk of invasion. (ii) A chart view, show either the temporal change or the categorization of the capture records by some attributes. (iii) An environment view (Fig. 2.12), illustrating the relationship between the time of year and the number of bumblebees caught on three species of flowers. The color represents the different years and the radius of the circles represents the number of capture records reported on each day for each flower. However limitations can be found on these views, besides Google Maps built in interactive mechanisms on the map view, no more means of interactivity is presented. The use of 3D bars offers occlusion and hard comparisons between bars. On the environmental view visual cluttering and overlapping is visible and comparisons are also hard to make.

![Figure 2.9: Ferreira et al. [5], system overview.](image)

Figure 2.9: Ferreira et al. [5], system overview.

![Figure 2.10: Kitamoto et al. [6], the map view of capture records in 2008. The height of bars represents the number of bumblebees caught by participants at one place through the year. Users can zoom in or out to see the exact location, and the clicking of bars shows the list of records in an info window.](image)

Figure 2.10: Kitamoto et al. [6], the map view of capture records in 2008. The height of bars represents the number of bumblebees caught by participants at one place through the year. Users can zoom in or out to see the exact location, and the clicking of bars shows the list of records in an info window.
Figure 2.11: Kitamoto et al. [6], a warning map created from survey data. Red color represents grids of high colonization probability, where green color represents the grid of low colonization probability.

Figure 2.12: Kitamoto et al. [6], relationship between flowers and bumblebees.

2.2.4 Other approaches

Regarding animal movement data, in the research by Shamoun-Baranes et al. [41] a theoretical workshop brought together specialists from diverse areas to discuss and work together on establishing what a common framework for discussions and interactive data exploration should be. Four main topics were approached: uncertainty and granularity, scalability, space and time and segmentation and clustering. The group of specialist debated and shared their thoughts on how to approach each issue and what techniques should be considered, however they don’t apply it in a practical way.

2.2.5 Visualization Discussion

Information visualization techniques gives researchers opportunity to take insights in an interactive, faster and better way than ever before. Despite all the different methods used in the referenced works, geographical distribution visualizations shows themselves the most popular ones, in which most, except Ferreira et al. [5], take advantage of the already existent Google Maps mechanism.
and all its relishes to build their visualizations on top of it. As Table 2 shows the use of linked views among the studied approaches is lacking, and although most of the works use multiple views, they are rendered on top of the previous one except [3, 5] providing an opportunity to improve on such limitations.

Table 2.2: Visualization studied approaches overall characteristics

<table>
<thead>
<tr>
<th>Studied approaches</th>
<th>Focus + Context</th>
<th>Multiple Views</th>
<th>Linked Views</th>
<th>Tooltips</th>
<th>Zoom + Panning</th>
<th>Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moustahfid and Weise. [1]</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Halpin et al. [2]</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Zheng et al. [3]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Silvestro et al. [4]</td>
<td>-</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>✓</td>
</tr>
<tr>
<td>Ferreira et al. [5]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Kitamoto et al. [6]</td>
<td>-</td>
<td>✓</td>
<td>-</td>
<td>-</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>Shamoun-Baranes et al. [41]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Hunter et al. [42]</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
# Recognizing Species

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Initial Idea</td>
<td>26</td>
</tr>
<tr>
<td>3.2</td>
<td>FishEye Recognition Algorithm</td>
<td>27</td>
</tr>
<tr>
<td>3.3</td>
<td>Sample Gathering and Treatment</td>
<td>28</td>
</tr>
<tr>
<td>3.4</td>
<td>Training</td>
<td>28</td>
</tr>
<tr>
<td>3.5</td>
<td>Detecting and Classifying</td>
<td>29</td>
</tr>
<tr>
<td>3.6</td>
<td>Architecture</td>
<td>29</td>
</tr>
<tr>
<td>3.7</td>
<td>Evaluation</td>
<td>30</td>
</tr>
</tbody>
</table>
A species’ recognition module was incorporated within FishEye’s functionalities taking into account possible usage scenarios, for instance: a user is visiting a pet shop and wants to know more about a particular fish species. If he knows the fish species’ name he can then use FishEye to further learn about such species, at that particular instant or at a later moment. However, if he doesn’t know the species’ name, he can take a photograph of the fish and use FishEye’s to recognize and identify such species’ name, then also choosing to visualize such species’ meaningful information at the moment or at a later time period.

The recognition process can be divided in two major phases, feature extraction in which fish species unique characteristics that differentiate them among each other are collected, and fish classification, that uses the gathered features and compares them with a knowledge base of previously gathered features trying to find a match, and so being able to efficiently classify that particular fish specimen.

The recognition development process was made following an iterative approach. The first step of such process was to find a library for image processing that could be used as backbone and which could support the real-time response speed that FishEye had as goal. After some research, OpenCV[^1] a reliable tool and library containing algorithms for image processing proved itself as a valuable framework for the intended objectives, as also supported the majority of the previously studied techniques.

As proof of concept for the provided recognition module, the Clownfish (*Amphiprion Percula*) species (Fig. 3.1), and Powderblue Surgeonfish (*Acanthurus leucosternon*) species (Fig. 3.2) were chosen.

![Figure 3.1: Clownfish (*Amphiprion Percula*) species.](image1)

![Figure 3.2: Powderblue Surgeonfish (*Acanthurus leucosternon*) species.](image2)

### 3.1 Initial Idea

In the early stage of development the main idea for the recognition process was to first segment the fish shape from the picture, in an automatic manner without user interaction, and only after feature extraction and classification processes would then be applied to the segmented shape.

By doing the segmentation first, noise and redundant data present on the picture would then be eliminated and following techniques would be done only on the fish itself greatly potentiating the quality and performance of such methods.

Several segmentation techniques were tried out such as edge detection algorithms like the Canny

[^1]: [http://www.opencv.org](http://www.opencv.org)
edge detector [44], thresholding [45], contours detection [46], blob [47] and shape detection [48]. However, due to several variant factors that can be found from picture to picture such as illumination, background, water color, fish shape, position, scale, among others, the proposed automatic segmentation process was found itself to be too complex and hard to be achieved in a global manner, such as FishEye intended to be.

Due to this hindrance, alternative methodologies were then studied and taken into consideration, one of those was the Object Detection [49] methodologies that OpenCV supported.

### 3.2 FishEye Recognition Algorithm

Based on initial exploration, object detection using Haar feature based cascade classifiers [50] proved itself as an effective object detection algorithm and chosen as the main algorithm for FishEye recognition process.

The work-flow for the algorithm is as follows: first a database of positive samples (images containing the fish) (Fig. 3.3), and negative samples (images containing common background for fish such as coral, sand, rock and water) (Fig. 3.4) is created. Based on these positive and negative samples, the next step is then training the classifier. Afterwards, the classifier is then ready to be applied to any given image input detecting a fish species efficiently.

![Figure 3.3: Some Clownfish positive samples.](image-url)
3.3 Sample Gathering and Treatment

The sample gathering process was done using several YouTube videos that included each species. By using different videos, variance of the same factors that affected the previously explained segmentation processes were assured to be taken into account.

In order to automate sample collecting for each video, a script was created by using Processing, an open source computer programming language for image processing. The script saved a frame every two seconds followed by resizing it to a much smaller and uniform size and applying a noise reduction treatment such as Gaussian blurring and histogram equalization. Afterwards, each frame was manually cropped to around two to four samples: one containing only the region of interest - i.e., the fish, and the remaining serving as background samples.

At present, FishEye’s sample database consists of a total of around 2,000 positive samples (half for each species) and roughly 2,000 samples for negative samples.

3.4 Training

The training stage starts with the feature extraction from gathered samples on the previous step, these features also known as Haar features, include edge, line and four-rectangle features to be analyzed and extracted.

The training process had many possible configurations, so to achieve the best possible within the given dataset, several were tried and compared against each other, so that FishEye uses the best possible.

---

2 https://www.youtube.com
3 https://processing.org
3.5 Detecting and Classifying

To classify a given image, it is divided into several sub-segments. The several features extracted are also divided hierarchically, grouped into 18 stages of classification, each increasing rigorousness. For each image sub-segment, the classification is an iterative process, being fed to each stage of the classifier, if it fails one stage then that particular sub-segment is discarded.

A detected fish shape is then considered the largest group of sub-segments that passes all the classifying stages.

3.6 Architecture

FishEye’s species’ recognition’s module architecture (Fig. 3.5) consists of three major components: FrontEnd, BackEnd and Database.

Users only interact within the FrontEnd component on which they upload the fish species’ picture they want to identify. The picture is then submitted by means of a communication protocol to the BackEnd component. On this component the pictures gets its features extracted and fed to the classification stage, where such features are compared within FishEye’s existing ones, stored in a knowledge base (a multimedia database where for each fish species their extracted features on the training/testing phase were recorded), by means of the classification techniques. After recognizing the fish species, the user is then redirected to the respective species’ information visualization page, which has a welcoming screen showing the recognized species’ basic info (Fig. 3.6) and prompting the user to know more about such species.
3.7 Evaluation

FishEye’s evaluation process was done regarding two fronts, a try and compare methodology during development and at a final stage a more rigorous evaluation following the same principles and metrics used across studied related approaches.

During development, since the sample gathering was a time-consuming and exhaustive process, every 250 samples, a dataset of ten pictures per species was fed to the trained classifier, having in mind that these dataset pictures were independent from the one used at the training stage. The main metrics of observation to validate the performance of each iteration of evaluation were the time lapse duration from inputing until a result was given, the number of sub-segments which the classifier identified as being fish, being considered the biggest segment area as the one with more probability of containing the actual fish and plotted in the picture. As example in the following table 3.1, a comparative between the picture with the worst metrics at the 250 samples iteration and the last iteration with around 1.000 samples is presented.

<table>
<thead>
<tr>
<th>Samples number</th>
<th>Time (s)</th>
<th>Number of sub-segments</th>
<th>% of Fish area detected</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>5.2</td>
<td>7</td>
<td>10 %</td>
</tr>
<tr>
<td>1.000</td>
<td>1.6</td>
<td>1</td>
<td>90 %</td>
</tr>
</tbody>
</table>

As represented in table 3.1 all metrics had a great improvement. To better achieve a state of representative improvement between these two iterations, the percentage of fish area detected is plotted in the following figures 3.7 and 3.8.
As mentioned previously, after the last sample gathering and training iteration leading to around 1,000 samples per species a more rigorous evaluation was done taking into consideration the same evaluation methodology than the studied related solutions.

To perform the evaluation, different datasets were used than those of the previous training phase. After, 50 samples for each species were randomly collected from these datasets, for which the evaluation of FishEye classifier was done.

For each sample testing, the following metrics were collected: true positive results (species identified correctly as depicted in Fig. 3.9), false positive results (species identified wrongly), false negative results (no species identified when it should), Fig. 3.10 and time duration since providing a particular
species sample until the getting of a result.

Figure 3.9: Clownfish correctly identified.

Figure 3.10: Clownfish not identified.

The gathered metrics allowed to calculate the following equations: Average Recall (AR), Eq. \( \text{AR} \), Average Precision (AP) Eq. \( \text{AP} \), Accuracy over Count (AC), \( \text{AC} \) and Average Response Duration (ARD), \( \text{ARD} \). These equations allowed to rate the performance of FishEye’s recognition process.

\[
\text{AR} = \frac{1}{\#\text{species}} \sum_{i=1}^{\#\text{species}} \frac{\text{TruePositives}_i}{\text{TruePositives}_i + \text{FalseNegatives}_i}. \tag{3.1}
\]

\[
\text{AP} = \frac{1}{\#\text{species}} \sum_{i=1}^{\#\text{species}} \frac{\text{TruePositives}_i}{\text{TruePositives}_i + \text{FalsePositives}_i}. \tag{3.2}
\]

\[
\text{AC} = \frac{\sum_{i=1}^{\#\text{species}} \text{TruePositives}_i}{\sum_{i=1}^{\#\text{species}} \frac{\text{TruePositives}_i}{\text{TruePositives}_i + \text{FalsePositives}_i}}. \tag{3.3}
\]
\[ \text{ARD} = \frac{1}{\text{#species}} \sum_{i=1}^{\text{#species}} \text{TimeDuration}_i. \]  

(3.4)

After applying the described equations to the collected metrics, the results achieved by FishEye were as follows:

<table>
<thead>
<tr>
<th>AR</th>
<th>AP</th>
<th>AC</th>
<th>ARD</th>
</tr>
</thead>
<tbody>
<tr>
<td>95%</td>
<td>100%</td>
<td>100%</td>
<td>1.61 s</td>
</tr>
</tbody>
</table>

Table 3.2: FishEye evaluation results

The achieved results shows that FishEye can efficiently recognize the trained species. For the 100 tested samples, only two regarding the Clownfish species and three regarding the Surgeonfish Species could not be identified. This leads FishEye to show an overall accuracy of 95%. Given that during the tests there was no occurrence of a false positive result, i.e., a species was never identified erroneously, this leads to 100% precision as well as a 100% accuracy for identified species. The average response time for the set of samples tested was 1.61 seconds which shows that FishEye can have a real-time response time as was its goal.

In order to improve these results, it would be enough to continuously train the classifier, thus improving response time as well as overall accuracy. In order to consolidate the obtained results, new species should be trained and tested, being this process independent, since new species can be easily added in the architecture of FishEye, as plug-ins.
4

Visualizing Species’ Information

4.1 Data .................................................. 35
4.2 Approach ........................................... 37
4.3 Architecture ....................................... 56
4.4 Demonstration of Potential .................... 57
Nowadays, a lot of information about fish species can be found, due to the continuous improvements across animal tracking technologies. However such data, presents itself as heterogeneous, disorganized (fragmented), difficult to relate to and have an overall view. This problem, so frequent today, due to the ease of generating and storing info, is often solved through Information visualization (InfoVis) techniques.

Information Visualization alleviates cognitive load associated with data interpretation [9]. A meaningful visualization allows in fact the representation of information in a way that highlights relevant information while providing the means for the user to explore and find important patterns. Represented data can be used to explore the intrinsic and external factors that may influence animal behavior, and so obtain a better understanding of the ecosystem function and evolutionary constraints of species.

FishEye’s visualization development followed an iterative and incremental paradigm. The following sub-chapters describe the several phases of the work done, starting by presenting the data attributes represented among the visualization giving a meaningful display of information to users, followed by describing the several idioms’ creative process and choices made across development. To validate such choices, an expert evaluation with a set of heuristics suited for information visualization [10] was performed, whose results led to FishEye’s improvement.

4.1 Data

The available data presented on the current solution is available at GBIF (Global Biodiversity Information Facility) [7], a free and open repository which gathers data from museums and scientific researches worldwide.

For each species, the available file at GBIF was downloaded, which was in .csv format. A sample of the information displayed on the files is depicted in Fig. 4.1. As can be observed, there is a lot of meta-data and scattered data columns, such as several ids, keys and codes, authorship and license, among others, which don’t represent meaningful information to be visualize in the context of our work. Nevertheless, some useful information can be retrieved, such as geographic and temporal information about each occurrence. Furthermore there’s also information about the depth at which the species were observed. Although a column with elevation information at which the species were collected exists, no record was found to have this column filled.
In order to obtain only information considered relevant and necessary within the context of our work, such as the attributes described previously, Pentaho Data Integration [51] software was used. The software allowed the removal of unwanted columns, the merging of files, as also the mapping of each country column to its belonging continent.

After treating the data, the dataset became much cleaner, containing only relevant information, with an average of 250 registries per fish species. Each species’ record consists of the species’ name, temporal information on month and year in which it was collected, geographical information such as GPS coordinates, continent and country as also the depth (in centimeters) at which the species was found (Fig. 4.2).

Furthermore, physical characteristics data was also collected in order to complement the existing dataset as also creating richer scenarios of interaction and learning about the species. Such information was collected from FishBase [52] and stored in a proper .csv file, which included information about some fish traits such as the number of dorsal spines, anal spines, dorsal soft spines, anal soft spines and common length (in centimeters), Fig 4.3.

<table>
<thead>
<tr>
<th>Species</th>
<th>Length</th>
<th>Dorsal_Spines</th>
<th>Dorsal_Soft_Anal_Spines</th>
<th>Anal_Soft</th>
</tr>
</thead>
<tbody>
<tr>
<td>clownfish</td>
<td>11</td>
<td>9</td>
<td>15</td>
<td>2</td>
</tr>
<tr>
<td>surgeon</td>
<td>19</td>
<td>9</td>
<td>30</td>
<td>3</td>
</tr>
</tbody>
</table>

Having the data we needed, properly processed, before moving on to the development process, the best approach in order to represent our data in the most meaningful way possible was discussed.
It also was taken into account the underlying characteristics of our data (temporal, geographic and species’ physical traits).

### 4.2 Approach

FishEye’s development process followed an incremental and iterative approach. The development process started by brainstorming ideas based on the available data. Furthermore, to a better understanding on what users think and want to know more about when visiting a museum, a partnership with *Aquário Vasco da Gama*\(^1\) was established. The partnership did not go forward, however the feedback and discussed topics within the meeting held were nevertheless valuable and enhanced the potential of FishEye. Having all the gathered knowledge so far into consideration the tasks the solution would need to support in order to answer meaningful questions were defined as follows:

- **Navigate** - Navigate through data across the years.
- **Identify** - Identify in which year/season there are more occurrences.
- **Locate** - Locate species’ countries and regions of election.
- **Explore** - Exploring how species occurrences change over time and across geographic regions.
- **Compare** - Compare different species.

To give context and meaning to the adjacent ideas Low Fidelity Prototypes (LFP) were used, followed by the use of Functional Prototypes (FP) on which such ideas were incrementally implemented, and afterwards, evaluated by means of an heuristic evaluation, thus leading to a more concise and robust final solution: *FishEye*.

#### 4.2.1 Low Fidelity Prototyping (LFP)

Low Fidelity Prototypes helped to conceptualize several visualizations based on initial ideas. These had in consideration the nature of the available data attributes. Regarding temporal data, the idea was to visualize time in a cyclical manner and so giving the possibility of identifying patterns as also extrapolating forecasts. To apply the previous idea, a *Radial Chart* was chosen as a mean to analyze the periodicity of occurrences, throughout each year, making seasonal events evident (e.g., were there more measured occurrences of Clownfish in Australia during summer months?). The circular layout provided the intended cyclical behavior, being divided in several rings, one for each year, and each ring divided in twelve segments representing corresponding months (Fig. 4.4). Color coding is used to represent the intensity of occurrences, directly mapped with the brightness of the chosen color.

\(^1\) [http://ccm.marinha.pt/pt/aquariovgama](http://ccm.marinha.pt/pt/aquariovgama)
In relation to data of geographical nature, the main idea was to help understand species’ population evolution over time and migration patterns. To achieve such goal a Mercator Projection Map was chosen. Each species’ occurrence is then plotted by means of circles, precisely mapped according to the respective GPS coordinates attribute, depicted on Fig. left side [4.5]. As can be seen on the figure, zoom and panning mechanisms are available in order to navigate through the map. Also, on the right side of the figure, there is a simple Line Chart meant to display depth evolution throughout time.

For a better sense of how many occurrences are distributed among respective countries, a Radial Bar Chart was chosen, facilitating the identification of species’ top countries on which they can be found. (Fig. 4.6).
The final sketched prototypes were made regarding categorical attributes, such as species' physical aspects (number of dorsal rays, dorsal spines, anal rays, anal spines and species' common length). A Radar Chart was chosen to display this information, since it provides a good way of displaying multivariate data and also allows easy comparison between multiple species (Fig. 4.7).

Some of these characteristics are unknown to the common person, so as complementary information and for a sense of awareness a generic fish shape is used to illustrate where such regions are located as depicted in Fig. 4.8.
During the process, it also was considered interesting to visualize some key information about the species, such as climate preferences and reproducing nature. For this purpose a Bubble Word Cloud visualization was chosen, mapping each bubble size to the keyword’s prominence/relevance respectively as represented in Fig. 4.9.
4.2.2 Functional Prototyping (FP)

Having a general notion of the graphical aspect of each intended visualization, the process of implementation for such took place. Each visualization was at a first stage individually implemented and later on incorporated together on a single page.

The technologies used at this stage of development were then: HTML5 for the page's construction and CSS3 for its styling, Bootstrap for the responsive page layout, jQuery and JavaScript for overall interactivity across the page and finnaly, D3.js a library for manipulating documents based on data [53] for constructing the visualizations.

Concerning the Radial Chart visualization, before implementation, it came to attention that instead of a radial layout, a spiral layout would represent better the intrinsic time cyclical behavior, having a starting point (center of the spiral) "unrolling" towards the most distant point, which couldn’t fully be displayed by the radial layout. Having this in consideration, the visualization layout changed to a spiral’s (Fig. 4.10). Despite the change, described characteristics for this visualization remained intact.

![Figure 4.10: Functional Prototype - Spiral visualization representing time cyclical behavior.](image)

As new iterations of development went by, improvements and new features were implemented (Fig. 4.10). The first improvement was to choose a better color coding: a pastel blue color was chosen, provided by Color Brewer [54], a widely used and online tool which provides good color scheme’s to
use in graphic visualizations based on color principles, number of data classes and the nature of the data (sequential, diverging and qualitative) [55]. To give meaning and be able to understand the values coded by the displayed color hue, a label scale was added with such information. The displayed data's temporal interval was also added by having in the spiral's middle section the corresponding starting year and at the finishing point the corresponding finishing year. As can be observed in the initial spiral display in Fig. 4.10 the spiral's direction was counterclockwise, which was counterintuitive. With this in mind, corrections were applied and the spiral's direction changed to clock-wise, furthermore, using the clock-numbers as metaphor, the month labels were organized accordingly.

A few filtering mechanisms were also added, such as the ability to disable/enable months, removing/adding a month's respective data, by clicking on its label. The possibility to change the years' interval on the displayed data was also added by means of an horizontal range slider. The slider also helped identifying the spiral's segments corresponding year, by using a scale as layout. Nevertheless, for better identification purposes of the segment's coded information (year, as also the corresponding value) tooltips were added, triggered by hovering over them. A detail-on-demand mechanism was also added, in order to get a better look at how occurrences were distributed along a particular year by using a simple Line Chart, triggered by clicking on any segment.

**Figure 4.11:** Functional Prototype - Spiral visualization representing time cyclical behavior with further developments.
Regarding the Mercator Projection Map visualization, the D3.js library provided the structure for the map itself. For a better looking and professional layout, a graticule and the equator line were added on top of the map. The next step was the mapping of the actual data GPS coordinates attribute. These was achieved by dividing the map into four equal parts, like a simple XY graph and projecting the coordinates accordingly, taking in consideration width and height.

One major change at this stage of development was to link the depth attribute, which was previously display on a simple line chart, with this visualization. By using color coding on the plotted circles, depth information at time of collection for each occurrence was made available. The color chosen to represent this information was green, since according to Ware and Kaufman [56] humans can perceive shades of green more readily than any other color because of the combined color perception of rods and cones as also from an evolutionary point of view. Lighter colors were use to identify lower depths and darker colors deeper ones. This led to a deeper understanding on the species’ behavior and patterns, for instance one can wonder for a particular region if the depth of occurrences remains the same all year long, or even in a decade period.

As filtering/interactive mechanisms, zooming and panning across the view were made available, being the zoom mechanism implemented as semantic zoom, maintaining circles’ size consistency across all zoom levels. These mechanisms are enhanced by using transitions easing user into changes.

Such as with the spiral visualization, in order to understand the value behind the used color coding, a legend scale was added on the view’s bottom left side.

All the described features can be observed in Fig. 4.12.

![Figure 4.12: Functional Prototype - Mercator Projection Map showing species’ distribution.](image)

Still in the context of geographic visualizations, the Radial Bar Chart displaying occurrences by country distribution was implemented as depicted on its correspondent low fidelity prototype, represented in Fig. 4.13 on the left side. Due to the categorical nature of such representation, a proper color schema was taken from Color Brewer. A sorting mechanism was implemented in order to easily
Despite showing good results at early iterations, some issues came to attention as development continued. In fact, as more data was fed to this visualization, scaling and legibility problems became clearer. To solve the adjacent problem a Treemap idiom was chosen to replace the previous visualization. By using a treemap, a better use of useful space was assured, as also further interaction mechanisms. Due to treemap's hierarchic nature, displayed information is at first related to the geographic distribution among world continents (Fig. 4.14 left side). Further detail about a specific continent is then made available by clicking on its respective bounding box, transitioning to the next level of hierarchy - the continent's belonging countries as depicted in Fig. 4.14 right side.

Each treemap's region has its area calculated according to the number of overall occurrences and ordered accordingly. Due to the limited size of some areas and for means of consistency, at the country hierarchic level, countries are identified by their respective International Organization for Standardization (ISO) code, however by clicking anywhere in the respective area, a informational tooltip with the country's flag and name as also the total number of occurrences registered is briefly displayed.

In order to navigate across the existing hierarchy levels, as also maintaining context on the displayed information, buttons following a breadcrumbs display were added on top of the view.
The last visualizations to be developed were regarding fish species’ characteristics, in fact only the Radar Chart and fish shape trait locations mapping were further developed. Due to constraints on the amount of visualizations that could be displayed on a single page without recurring to scroll mechanisms, a visualization was chosen to be discarded, and since minimal value was offered by the Word Bubble Cloud when compared with the remaining visualizations, it was discarded.

For a more interactive experience within the radar chart, this visualization was integrated together with the fish shape since both had common regions being displayed. By clicking on each available physical attribute on the radar chart, their counterparts are highlighted within the fish shape, while also showing its respective value.  

![Figure 4.15: Functional Prototype - Radar Chart visualization displaying fish species physical traits information.](image)

Having all visualizations independently prototyped in a functional manner, the next step was to
incorporate them all together as also to start developing linking and integration mechanisms between one another.

Due to the common geographic domain of both Map Projection’s and Treemap’s idioms, these were the first idioms to be integrated and linked.

The following interactions and linking between these views are then: (i) Clicking on a continent’s treemap bounding box, will then focus on that particular world region on the map view, by automatically zooming and panning to that respective area. (ii) at the countries hierarchic level, clicking on a particular country, will then briefly highlight its counterpart on the map view, by giving it the same color coding as presented on the country’s treemap bounding box. Occurrences (circles) referring to that country will also be highlighted.

The reverse situation is also available, in which clicking on any country in the Projection Map idiom, will trigger the treemap to show the respective hierarchic level in which that country is inserted into, the same brief highlight mechanism is also maintained.

A representation of feature (i) can be observed in Fig. [4.16] on which Indonesia was clicked on the treemap idiom, leading to its highlighting on the map view.

Figure 4.16: Functional Prototype - Treemap and Projection Map linking and interaction mechanisms example.

Despite time already being represented on the Spiral idiom, it came to attention that only time’s cyclical behavior was being represented. Furthermore, it was difficult to perceive how species occurrences were distributed across the given temporal domain. To solve such issue the solution was to embrace and represent time’s linear behavior as well.

The chosen idiom to represent the intended behavior was a BeeSwarm plot, a one-dimensional scatter plot, with closely-packed, non-overlapping points.

BeeSwarm’s main role was to give an overview on how the species’ occurrences were distributed across time, and at the same time due to the inherit characteristics of the idiom forming a “flow effect” between points.

The idiom was then implemented alongside the Projection Map and Treemap views as depicted in Fig. [4.17]. The displayed circles were mapped directly according to the the slider domain, serving this as axis for the displayed data and possibilitating the user to be able to identify temporal periods of larger or smaller magnitude of occurrences.
Each circle was given a color coding accordingly to its correspondent continent, the same used in the Treemap idiom. This allows the user to quickly infer the corresponding region for each circle, and have a better understanding of how time affected each continent's occurrences. The same behavior was guaranteed for when the user wants further detail on a continent, by clicking on it in the Treemap idiom. When the temporal slider is used, circles outside the chosen range are faded out and not completely erased, just to maintain the overall timeline display purposes intended for this visualization.

Furthermore, improvements were made at this stage. Regarding the Projection Map, a new legend scale was added. The new scale presented a continuous domain by means of a color gradient facilitating the interpretation of the meaning of a circle's color. An option to get the corresponding detail for each circle, similar to the already existing on the treemap, was also added displaying the correspondent country and depth information, triggered by clicking on a circle, also causing that particular depth to be highlighted on the legend scale.

As can be observed in figure 4.17, some circles present a purple color coding. Circles with such color coding represent data records in which no depth information is given, however since these items still represent a geographic occurrence, they are still plotted and accounted for. This information is also available in the legend scale. The chosen color (purple) was picked at ColorBrewer [54], taking into consideration the already at use color's codings among visualizations, as also contrasting well with the green color coding being used among the remaining circles.

At this iteration, the temporal slider was also given a design change, since in the previous design the existing buttons were too small, new buttons were made larger. Also, in a previous version the color green was being used as background for the slider, however this color was already widely used among visualizations and could mislead users on its interpretation.

![Figure 4.17: Functional Prototype - BeeSwarm implementation.](image)

With already three fully functional idioms, linked together and integrated with each others, the next step was to add the Spiral idiom.

Due to the current visualization's page continuous growing, we felt it was time to start creating a proper interface to contain the views. The interface design itself was kept simple and properly maximizing the useful space for the visualizations, due to the constraint of a non scrollable solution.
The chosen layout consisted of a navigation bar showing the current species on which information is being displayed as also the FishEye name and logo, being the rest of the page reserved for the visualizations. Each visualization is incorporated in a panel, which are collapsible allowing users to hide and show content as they please. The described interface, already containing the spiral view within, is depicted in Fig. 4.18. As can be observed in the figure, the spiral self-contained slider was removed, and its function delegated to the overall slider.

![Functional Prototype - FishEye’s first Interface layout.](image)

As the incremental development process continued, further functionalities were added. One of the primary concerns of the proposed solution was giving the user the possibility to re-arrange and resize the visualizations’ panels as needed, thus creating a more personal experience. With a simple drag-and-drop mechanism users can change the default dashboard layout. Also by means of resizing, the user can have a better look at the information. All these interactive mechanisms are responsive and automatically adapt themselves while maintaining the current information context. Regarding the visualizations, a few change were introduced to the Projection Map and Spiral idioms.

The Spiral visualization was given a new legend scale following the same principles of the already existing one on the Projection Map, maintaining the design’s consistency. The major change within
this view, was the month filtering mechanism improvements. Starting by merging the month labels text and buttons within a single element, providing a more aesthetic look. Also, instead of simply removing the elements of a particular month when chosen to be removed, leaving a white gap, now the spiral redraws itself in order to display only the enabled months, at the same time, maintaining a list of disabled months next to it on which the user can click to re-enable them. It’s important to note that, when any view is redrawn, it maintains all the active filters that were applied, never losing the active context of information.

In relation to the *Projection Map* idiom, the only significant change was placing the legend inside the map window, giving it more consistency and a better aesthetic feel.

All the above improvements can be visualized in Fig. 4.19 taking into consideration that the layout organization was made by means of the new interactive resize and moving features, also since no further changes were made to the *BeeSwarm* and *Treemap* idioms at this stage, they encounter themselves collapsed.

![Figure 4.19: Functional Prototype - FishEye's Interface layout improvements.](image)

The last visualization to be added within FishEye’s dashboard was regarding the species’ physical traits. Due to its simple complexity no further features were implemented for this idiom. In fact, the only change was to move the generic fish shape from below the *Radar chart* to its right side, for space saving reasons. Nevertheless at the interface global level, changes were made. The first big change was regarding the overall aspect of the interface. By giving the panel headers a darker gray background color a better contrast with the overall lighted color layout was created, which also helped drawing more attention to each panel’s content - the visualizations. Also white space among panels, although not all, was considerably removed. This was achieved by making the visualizations, namely the *Map Project* and *Treemap* idioms, use its entire panel’s size.

The panel in which the *BeeSwarm* idiom was inserted was removed. This decision was made in
order to differentiate this particular view from the remaining since it was meant to serve only as a road-
map across time within the displayed data, aiding users to have a bigger picture of the occurrences
distribution as also facilitating the identification of temporal slots of interest.

A time-lapse visualization option was also implemented. This feature allows the user to observe
how species occurrences evolved across all domains throughout time without further interaction with
the system. This feature is available through a play button above the temporal range slider. Clicking
on the button will reset the slider handles to the temporal starting point and as time increases one
of the handles will start increasing the timespan visualized until reaching the last available temporal
record. While the handle automatically moves along the slider, all visualizations adapt their informa-
tion properly, only containing data belonging to the current time interval. Users can stop or pause this
process at any time.

In order for users to better identify a year’s location throughout the slider a scale was added to
the slider axis. At this point, the existing buttons previously used to navigate through the Treemap’s
hierarchic levels, were in fact affecting all visualization, filtering the displayed data accordingly, so we
decided to display it on the navigation’s bar right corner, a more centralized point within the dashboard.

The last improvement at this iteration was regarding the Spiral's detail view. Previously, by clicking
on any segment, a line chart with the occurrences’ evolution throughout that segment's year would be
displayed. However, it was hard to perceive how many occurrences there were on each month. So a
bar chart idiom was used instead, while maintaining the intended occurrences’ evolution information,
as also, displaying the respective amount of occurrences for each month at the top of each bar. All the
described improvements are represented on Fig. 4.20, a case scenario in which the user clicked on a
2012’s spiral segment and information about that particular year was given across all visualizations.

Due to our process of incremental and iterative development and having all visualizations fully
functional as also linked together on a single page, with several interaction and filtering mechanisms across all views we decided to perform an heuristic evaluation in order to assess and validate the current status of our solution.

4.2.3 Heuristic Evaluation

Heuristic evaluation is essential as it provides a quick and low effort kind of evaluation, with meaningful results. This evaluation process was done with five experts with a background in the Information Visualization domain, during the mid cycle of development. In order to assess the solution, the experts were given a brief guide explaining FishEye context and each visualization’s purpose and features. The guide also contained a set of ten heuristics, which are properly indicated to be used within Information Visualization projects, based on the research by Forsell and Johanneson [10], on which the evaluation process would be based upon.

A total of 39 heuristics violations were identified by the experts. Each record had information on the name of the violated heuristics, the violation severity on a [0-4] scale, problem description and possible solution. After, all records were analyzed and a process of information consolidation took place on which similar problems were grouped together as one item, thus reducing the total of violated heuristics to 32 records. On table 4.1 and table 4.2 results on the identified violated heuristics can be found ordered by frequency and by severity.

### Table 4.1: Violated heuristics frequency

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation and help</td>
<td>8</td>
</tr>
<tr>
<td>Spatial organization</td>
<td>6</td>
</tr>
<tr>
<td>Information coding</td>
<td>5</td>
</tr>
<tr>
<td>Recognition rather than recall</td>
<td>4</td>
</tr>
<tr>
<td>Consistency</td>
<td>3</td>
</tr>
<tr>
<td>Data set reduction</td>
<td>3</td>
</tr>
<tr>
<td>Flexibility</td>
<td>3</td>
</tr>
<tr>
<td>Prompting</td>
<td>2</td>
</tr>
</tbody>
</table>

### Table 4.2: Violated heuristics severity frequency

<table>
<thead>
<tr>
<th>Severity</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

The main range of violated heuristics severities identified revealed some minor and major usability problems at the time being. These were mainly on how the user navigated through the solution and performed actions as also the overall layout chosen for the displayed information. To give some context to the experts’ identified problem, a description and possible solution for the top three violated heuristics is given as follow:
• Problem: "I couldn’t understand that clicking in the circles in the Map Projection or in the squares in the TreeMap was possible to obtain more information."

Solution: "Try changing the mouse when hovers the circle and squares to a hand symbol or something like that."

• Problem: "The body parts of the fish (Physical Characteristics view) get the same fill colors as the ones used to identify continents/countries on other idioms. This could make the user think there is a relation between the location names and the fish’s body parts."

Solution: "Try changing the mouse when hovers the circle and squares to a hand symbol or something like that."

• Problem: "One cannot identify what is being color coded at the projection map"

Solution: "Add a label on top of the existing legend, identifying it as the depth attribute."

The performed evaluation validated the performed work made so far, however it also revealed a considerable amount of problems regarding several domains. Having this in consideration, the development process continued, focusing first on the identified problems’ correction, which would then lead to a more high quality work and FishEye’s final version.

4.2.4 FishEye

After the performed heuristic evaluation FishEye’s incremental and iterative development process continued, focusing on improving the existing features, adding new ones and solving problems. All these, led to FishEye’s final version, depicted in Fig. 4.21.

As can be observed an uniform color coding was given to all idioms, using the same green color that was already present on the Map Projection idiom. This was done in order to maintain consistency
and uniformity across all views.

Regarding the Spiral visualization, beyond the new color coding, month labels were given a better color coding as also symbols indicating that they can be clicked on to remove any month. Empty space was also reduced, creating more useful space for the remaining visualizations. The spiral was added a responsive mechanism in order to always adapt to its panel maximum-width, as consequence for species whose data has a greater temporal range, segments are hard to perceive and interact with, so a zooming and panning mechanism was added in order to help users in such cases. Hovering mechanisms by means of tooltips were also implement in order to get further detail about each segment respective month, year and number of occurrences.

In respect to the Map projection idiom, some circles were overlapping others, so opacity mechanisms were implemented to solve such problem. Also as means of consistency across views the same zoom buttons as the ones implemented in the Spiral’s were added.

The Treemap view was also given minor improvements. For some cases, the computed areas were too small to be able to properly read their content, so to avoid errors and since detail mechanisms are at hand, text is not displayed for those cases.

4.2.5 Comparing Species

Species’ comparison across the existing domains provides further knowledge about them. Furthermore, it helps identifying and establishing relations between multiple species. Species’ behavior can also be affected by other’s within their habits of choice, for instance, does the appearance of a new species, in a specific geographical location, lead to a decrease of others? FishEye can be used to answer such question. Initially the dashboard presents information about a single species, however new species information can be added by selecting it from a list of available species (Fig. 4.22). One can also change species on a singular level, meaning just changing the current species’ data being displayed to the one selected.
Upon selection, the views readapt themselves to the new information. On Fig. 4.23 comparison between the Clownfish and Surgeonfish illustrates the changes made to each visualization in order to support the comparison mode.

On the Spiral visualization, it is possible to compare which species had more occurrences throughout the years, by means of color blending mechanisms. Being each species represented by different color codings, in order to properly perceive the resultant color and its meaning further research was made.

According to the research by Gama and Gonçalves [57] on human perception of color components’
relative amounts in blended colors, one of the color pairs humans can most correctly perceive is the (green, yellow) pair, which in equally amounts results in a lime color.

With this in mind, the yellow color coding was chosen to represent the second species, while the first one kept the green color as coding. This results in spiral’s segments whose color is closer to yellow representing more abundance for the second species and vice-versa. When equal number of occurrences between the species occur segments present the lime color coding. In order to complement the adjacent mechanism, a proper color legend scale was implemented. Color brightness is still used to represent the magnitude of occurrences. To represent the preponderance between species hue is used. Since the possible combinations of number of occurrences between species, fish’s avatars were included near the legend to aid on perceiving the possible resulting colors.

The following combinations are then contemplated within the legend:

- Many Clownfish (dark green) and many Surgeonfish (dark yellow), will lead to a dark lime color;
- Few Clownfish (light green) and few Surgeonfish (light yellow), will lead to a light lime color;
- Many Clownfish (dark green) and few Surgeonfish (light yellow), will lead to a greener color with intermediate brightness;
- Few Clownfish (light green) and many Surgeonfish (dark yellow), will lead to a yellower color with intermediate brightness;

The idiom used to perceive information about a segment’s corresponding year was also modified. Since we want to compare the evolution of both species across a particular year, a line chart (Fig. 4.24) idiom was chosen due to its simplicity. Each line being colored with its respective species’ color.

![Figure 4.24: FishEye’s species’s comparison mode regarding Spiral’s detail view.](image)

The remaining visualizations also went through some changes. The Treemap panel was divided in half, each representing one species, allowing the comparison between regions to be made side by side. Interacting within one treemap, will also trigger the other to display the same information.
On the Map visualization, new species are represented by appending new circles to the view, but with different color coding, the same yellow as in the spiral idiom for reasons of uniformness. Furthermore, a representative depth legend scale for the new species was also added.

Regarding the Radar chart view, a new area is plotted representing the new species attributes. A new species fish shape, representing the new species, is also drawn taking into consideration the length ratio between the two species and scaling it accordingly.

A new BeeSwarm was also added under the temporal range slider, maintaining the same temporal domain in order to properly compare occurrences between both BeeSwarms.

### 4.3 Architecture

FishEye’s visualization architecture can be divided into three major components, FrontEnd, BackEnd and Database. Users only interact within the FrontEnd component which is composed by three layers, the View Layer which is the actual page the user interacts with, the Styling Layer where all .css and media content is stored and the scripts layer, which stores FishEye’s idioms management and construction logic. To be able to construct the visualizations each script file needs data to be displayed. For each user’s interaction, the active visualizations does data requests to the BackEnd’s FrontEnd Communication Layer, this layer then delegates the data retrieval to the Database Communication Layer which identifies which data is needed for each request and requests it to the Database component, where all .csv files are stored.

After fetching the requested data, information flows backwards, from the Database component to the FrontEnd Component, in which the visualization is then created, applied the respective styling and finally rendered to the user’s visualization. The described architecture is represented in Fig. 4.25.
4.4 Demonstration of Potential

FishEye is a web application that helps users not only learning more about fish species, but also to be able to infer knowledge about it. By being deployed as a web application, FishEye is accessible to a wide range of devices and operating systems, without any kind of installation process, meaning it is accessible for any person. Due to its interactive mechanisms such as collapse, resize and drag-and-drop features, FishEye provides each user a personal and more richer experience, giving each user the possibility to create its own layout.

Several data domains are present in FishEye's available visualizations, such as temporal data (represented on both cyclical and linear manner) which helps identifying patterns and gives the possibility of extrapolating forecasts and geographical data which plays an important role when analyzing species' behavior, helping to understand the evolution of species' populations over time, migration and mating patterns as well as correlating one species' behavior with others. FishEye also educates users about species' physical characteristics and where they are located within.

FishEye also provides filters for its both geographical and temporal domains, allowing the user to visualize a large amount of data rearrangements, as also having more detail-on-demand for each idiom present, by click or hovering elements. With so many possible combinations it's important to maintain context among all views despite possible changes, which is also offered by FishEye. To further engage the user FishEye presents several animated transitions across all views in order to ease into changes.

FishEye’s also allows the display of information regarding two species at the same time, giving the
user the possibility of comparing species across all its existing domain as also establishing relations and common behaviors about such species.

FishEye’s visualizations empower and facilitate its users to easily gather knowledge about species, just by observing each idiom at hand. For instance, by observing Fig. 4.26 we can infer some conclusions about the Surgeonfish species’, such as:

1. Based on the BeeSwarm we can observe that, early occurrences were mainly registered across Africa, however, how progresses to the present, species’ occurrences can be found mainly in Asia, which strongly indicates species migrations across the two continents.

2. Based on the Map Projection it can be noted that the species’ habitats are mainly near shore, which also corroborates the maximum depth, at which Surgeonfish’s occurrences were found, being 20 centimeters deep.

3. Observing the Treemap idiom reveals the species can also be found in Oceania, which couldn’t be seen at first sight in the Map Projection.

4. Looking into the Radar chart, it shows the species’ average size is 19 centimeters and that it has many dorsal and anal soft rays. Furthermore, such rays location within the fish can also be observed.

5. Perceiving the Spiral it can be observed that, the species’ period of higher intense activity is found across February until May, while during Summer season few occurrences can be found.

In order to increase the scope of conclusions drawn about the Surgeonfish, it would suffice to use the available filters and also the detail on demand options present on each idiom.

The conclusions drawn go accordingly to the initially defined tasks which FishEye had to support.
Due to its architecture, FishEye can be easily added more species, just by adding the species’ information within the already existing database.
5 Evaluation

Contents

5.1 Preparation ......................................................... 61
5.2 Protocol .............................................................. 62
5.3 Results ................................................................. 62
5.4 Further Statistical Analysis ............................... 67
5.5 Discussion .............................................................. 69
After finishing the last iteration of development, a set of user tests was carried out in order to
gather quantitative usability metrics.
The main goal of this evaluation was to understand how, in a real life context, users would react and
experience FishEye. The evaluation stages were as follows: a preparation stage, where all necessary
materials were designed and created, the actual testing (following a well defined protocol), and after,
the analysis and discussion of the gathered results.

5.1 Preparation

The preparation process carried out led to the creation of two documents: a presentation script
used to introduce users to what FishEye consisted of, and what features it provided, and a questions
document, which users would have to answer while using FishEye.
By introducing users to the testing protocol and FishEye's itself using the presentation script, a coher-
ent and uniform discourse for all was ensured.

The questions document consisted of a set of twelve questions, with a varied range of complexities,
designed in such a way so that the user would have to interact with all features offered by FishEye,
thus having a full coverage of the application.
The set of twelve questions is presented as follows:

1. For the Clownfish, what are the months where there are more occurrences?
2. For the Clownfish, which continent has more occurrences?
3. What is the highest recorded depth for the Clownfish?
4. How many dorsal spines do Clownfish have in average?
5. In the 19th century, what were the continent and countries of choice for Clownfish?
6. How many occurrences of Clownfish in the Caribbean?
7. Comparing the Surgeonfish with the Clownfish, which one has a longer length?
8. On what month and year were recorded more occurrences of Surgeonfish, and how many?
9. Indicate a year where an equal number of occurrences were recorded for the Clownfish and
Surgeonfish?
10. For the Surgeonfish and Clownfish how many occurrences were recorded in Indonesia?
11. For the Surgeonfish and Clownfish, what species predominates on the African continent?
12. In the Asian continent, how it compares the evolution of occurrences along 2016 for both Sur-
geonfish and Clownfish?
5.2 Protocol

As a requirement for the evaluation process, it was established that it should be done with at least 20 users. No restrictions were created on the basis of age, gender or educational background, thus, reaching out to a wide range of profiles. It was also established that all subjects should have the same conditions of evaluation, hence a controlled and typical use context environment was used. The same tools to perform the tests were given among users.

Before the tests realization, a contextualisation on FishEye and the current evaluation goals was given to users. This consisted on following the beforehand prepared script described on the previous chapter, mainly giving an overall overview and explaining each visualization’s purpose and features.

Having finished the presentation, each participant was given a five minute period to explore FishEye as they pleased. Afterwards, participants were given the questions document and informed that for each question the same starting point would be used.

Subjects were informed that, for each question, while interacting with FishEye until an answer was given, an independent observer would collect metrics on time duration and number of errors made. Finally, users were ensured that the evaluation process was about testing FishEye and not themselves, giving them more confidence and comfort to freely explore the system.

After all twelve questions were answered, each subject was presented with the System Usability Scale (SUS) a simple, ten-item scale giving a global view of subjective assessments of usability [58] about the user experience with FishEye. Following the guidelines established by J. Brooke [58], the questionnaire took place right after each user finished the twelfth question without any debriefing or discussion. Users were also asked to answer each question with their first instinct instead of thinking much about it, and if undecided to pick the middle score of the presented scale.

Each item had a degree of disagreement or agreement, with a range from Strongly Disagree (1) to Strongly Agree (5) respectively, from which the user could choose.

5.3 Results

User evaluation was performed with 20 subjects. The age group between users was from 19 to 33 years old. These had a diverse range of educational qualifications in several engineering areas such as computer science, civil, electrotechnical, among others. As described on the previous chapter, while the user was answering each question, response time and number of errors made were annotated, and at the end the a System Usability Scale (SUS) questionnaire was answered.

5.3.1 Response Time

Regarding time duration, the results for each user and question are consolidated in Table 5.1 and for a better graphic display of such data, a representation by means of box-plot for the tabulated results was created, in Fig. 5.1 allowing us to observe where 50% of the most probable values, the median and the extreme values are located.
Table 5.1: Users tests duration for each question.

<table>
<thead>
<tr>
<th>User</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
<th>Q11</th>
<th>Q12</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>13.4</td>
<td>10.1</td>
<td>15.4</td>
<td>5.1</td>
<td>23.0</td>
<td>12.5</td>
<td>14</td>
<td>66</td>
<td>46</td>
<td>32.8</td>
<td>11.5</td>
<td>70.2</td>
</tr>
<tr>
<td>U2</td>
<td>5.6</td>
<td>1.2</td>
<td>0.9</td>
<td>6.6</td>
<td>32</td>
<td>31.5</td>
<td>12</td>
<td>32.5</td>
<td>53</td>
<td>54.4</td>
<td>11.6</td>
<td>26.7</td>
</tr>
<tr>
<td>U3</td>
<td>0.8</td>
<td>0.2</td>
<td>21.5</td>
<td>8.2</td>
<td>31</td>
<td>16</td>
<td>7.1</td>
<td>6.8</td>
<td>8.4</td>
<td>15.3</td>
<td>13.4</td>
<td>32.4</td>
</tr>
<tr>
<td>U4</td>
<td>18.4</td>
<td>0.7</td>
<td>16.6</td>
<td>6.7</td>
<td>16.1</td>
<td>11.2</td>
<td>9.7</td>
<td>31.9</td>
<td>13.4</td>
<td>10.8</td>
<td>11.6</td>
<td>30.5</td>
</tr>
<tr>
<td>U5</td>
<td>7.2</td>
<td>2.4</td>
<td>2</td>
<td>8.7</td>
<td>19.3</td>
<td>7.5</td>
<td>11.6</td>
<td>18.1</td>
<td>20.3</td>
<td>12.5</td>
<td>5.3</td>
<td>29.5</td>
</tr>
<tr>
<td>U6</td>
<td>9.4</td>
<td>3.3</td>
<td>10.3</td>
<td>9.2</td>
<td>24.3</td>
<td>16.8</td>
<td>5.8</td>
<td>22</td>
<td>44</td>
<td>21.4</td>
<td>7</td>
<td>47.2</td>
</tr>
<tr>
<td>U7</td>
<td>12.9</td>
<td>7.6</td>
<td>12.4</td>
<td>4</td>
<td>23.1</td>
<td>11</td>
<td>5.9</td>
<td>44.5</td>
<td>48.6</td>
<td>41.6</td>
<td>10</td>
<td>33.9</td>
</tr>
<tr>
<td>U8</td>
<td>1.2</td>
<td>1.1</td>
<td>3</td>
<td>5.7</td>
<td>7.4</td>
<td>23.1</td>
<td>8.3</td>
<td>12.4</td>
<td>27.1</td>
<td>39</td>
<td>11.5</td>
<td>35</td>
</tr>
<tr>
<td>U9</td>
<td>15</td>
<td>10.3</td>
<td>2.1</td>
<td>3.3</td>
<td>25.2</td>
<td>29.1</td>
<td>14</td>
<td>8.7</td>
<td>23.1</td>
<td>44</td>
<td>14.1</td>
<td>27.6</td>
</tr>
<tr>
<td>U10</td>
<td>1.2</td>
<td>0.9</td>
<td>12.3</td>
<td>5.3</td>
<td>20.5</td>
<td>13.1</td>
<td>9.2</td>
<td>14</td>
<td>23.4</td>
<td>19.2</td>
<td>9</td>
<td>18</td>
</tr>
<tr>
<td>U11</td>
<td>1.8</td>
<td>1</td>
<td>4.1</td>
<td>14.2</td>
<td>12.6</td>
<td>15.1</td>
<td>10</td>
<td>13.8</td>
<td>11.2</td>
<td>32</td>
<td>12.5</td>
<td>44.8</td>
</tr>
<tr>
<td>U12</td>
<td>4.4</td>
<td>3</td>
<td>10.2</td>
<td>6.1</td>
<td>11.1</td>
<td>19.4</td>
<td>15.2</td>
<td>32.1</td>
<td>14</td>
<td>17.9</td>
<td>8.2</td>
<td>25.7</td>
</tr>
<tr>
<td>U13</td>
<td>7.5</td>
<td>2.4</td>
<td>7.5</td>
<td>9.9</td>
<td>9.4</td>
<td>10</td>
<td>9.4</td>
<td>24.1</td>
<td>25.5</td>
<td>12.5</td>
<td>9.5</td>
<td>35</td>
</tr>
<tr>
<td>U14</td>
<td>0.9</td>
<td>1.7</td>
<td>1.4</td>
<td>12.6</td>
<td>22.7</td>
<td>19.4</td>
<td>8.6</td>
<td>9.2</td>
<td>15.7</td>
<td>9.5</td>
<td>13.1</td>
<td>42.4</td>
</tr>
<tr>
<td>U15</td>
<td>1.3</td>
<td>8.2</td>
<td>9.3</td>
<td>11.2</td>
<td>18.4</td>
<td>8.8</td>
<td>11</td>
<td>19.5</td>
<td>25.9</td>
<td>21.4</td>
<td>6.8</td>
<td>21.6</td>
</tr>
<tr>
<td>U16</td>
<td>16.2</td>
<td>5.4</td>
<td>7.2</td>
<td>5.6</td>
<td>32.7</td>
<td>14.5</td>
<td>14.1</td>
<td>43</td>
<td>41.5</td>
<td>32.7</td>
<td>11.3</td>
<td>60.2</td>
</tr>
<tr>
<td>U17</td>
<td>7.8</td>
<td>1</td>
<td>6.5</td>
<td>9.6</td>
<td>23</td>
<td>9</td>
<td>5.7</td>
<td>8</td>
<td>15</td>
<td>11.2</td>
<td>13.1</td>
<td>41</td>
</tr>
<tr>
<td>U18</td>
<td>15</td>
<td>9.1</td>
<td>12.4</td>
<td>11.3</td>
<td>12.1</td>
<td>21.4</td>
<td>6.6</td>
<td>16.8</td>
<td>23.4</td>
<td>19</td>
<td>7.3</td>
<td>29.4</td>
</tr>
<tr>
<td>U19</td>
<td>6.2</td>
<td>5.5</td>
<td>1.8</td>
<td>14</td>
<td>15.9</td>
<td>30.2</td>
<td>9.1</td>
<td>21.5</td>
<td>11.2</td>
<td>26</td>
<td>4.10</td>
<td>45.6</td>
</tr>
<tr>
<td>U20</td>
<td>3.2</td>
<td>8.3</td>
<td>29.1</td>
<td>8</td>
<td>12.8</td>
<td>9</td>
<td>15</td>
<td>40</td>
<td>26.9</td>
<td>31.1</td>
<td>3.2</td>
<td>37.1</td>
</tr>
</tbody>
</table>

Statistics

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>Mean</th>
<th>Max</th>
<th>Standard Deviation</th>
<th>Confidence Interval (95%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>0.80</td>
<td>7.47</td>
<td>18.40</td>
<td>5.83</td>
<td>2.55</td>
</tr>
<tr>
<td>Mean</td>
<td>0.2</td>
<td>4.17</td>
<td>10.3</td>
<td>3.52</td>
<td>1.54</td>
</tr>
<tr>
<td>Max</td>
<td>0.9</td>
<td>9.3</td>
<td>29.1</td>
<td>7.39</td>
<td>3.23</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>3.3</td>
<td>26.3</td>
<td>14.2</td>
<td>3.02</td>
<td>1.40</td>
</tr>
<tr>
<td>Confidence Interval (95%)</td>
<td>8.57</td>
<td>19.63</td>
<td>32.7</td>
<td>6.71</td>
<td>3.27</td>
</tr>
</tbody>
</table>

Figure 5.1: Box Plot for Question solving duration (s).
By observing both table 5.1 and the box-plot Fig. 5.1 some conclusions can be drawn.

Questions Q4, Q7, and Q11 present the smallest value dispersion, respectively, showing users have a high level of agreement with each other. The overall response for these questions can be considered fast with means of 8.26, 10.11 and 9.7 seconds, respectively. These results go according to expected, since these questions could be answered with minimal interaction, relying more on observation skills. The fourth and seventh questions, were relative alike, aiming to test the species physical characteristics visualization. However, users took a short extra time to answer Q7, this can be explained by the fact the question need for further interaction to be answered. The users need to use FishEye species comparison mode, as also, the fact this question represents the first question regarding species comparison, explains such results. Despite this, the mean of Q7 was much closer to its lowest box extreme, and so we can assume the user improved since the last interaction (within that particular visualization) at question four.

Questions Q1, Q2 and Q3 represent the answers with lowest minimum and mean response times, with a few response times under 2 seconds, even under 1 second for Q1 and Q2, proving these answers were quite intuitive for some users.

Q1 presents some considerable dispersion between answers, with standard deviation of 5.83 seconds. This can be explained since to answer the question users would need to interact with the Spiral idiom, being an unfamiliar way of representing the cyclical behavior of time for some users, the highest value registered was 18.4 seconds which shows that even for users unfamiliar with such data representation, they were able to interpret it in considerable time. Nevertheless, by observing the mean on Q1 box, a close to symmetric distribution can be observed.

Question Q2 is the one in which users had the fastest responses, with a small standard deviation of 3.52 seconds and a mean of 4.17 seconds close to the box bottom. Results above the mean can be explained by users trying to to answer this question by interacting and observing the Projection World Map visualization, while the rest answered recurring to the Treemap visualization which allowed to give a more intuitive and immediate response.

Question Q3, considered as having the same low complexity as Q2, on which the user needed to see the legend bar within the Projection World Map visualization, has a considerable level of result dispersion among users, with a standard deviation value of 9.3 seconds, more than double of Q2’s. The box’s top whisker also shows a maximum value of 29.1 seconds, a high value than Q1 and Q2 max values combined, showing the majority of users struggled with this question, as a matter of fact, some users were scanning darker circles on the map looking for the answer, when the answer was one inch away, showing the legend for the map needs some improvement on legibility and representation meaning.

Questions Q5 and Q6 were considered medium complexity, and as it can be observed they present a similar representation on the box-plot, which presents a medium-high values dispersion. These questions presented the user the first use of filtering mechanism and also zooming and panning through the Map Projection visualization, which can explain such dispersion.

Questions Q8, Q9 and Q10 can be observed as the ones with biggest values dispersion and stan-
standard deviation among the twelve questions set. Due to the fact that these questions were considered high complexity questions, having to change or compare species, as also, there were different ways and orders getting the correct answer, explaining such visible dispersion. Similar means and box sizes shows user consistency among these three questions.

The eighth question presented the biggest standard deviation value registered among questions with a value of 15.32 seconds, bigger then the slowest answer in Q2, Q3, Q7 and Q11. Together with the previously described factors, this question goal was to test if the user could interpret the blending mechanism present on the Spiral Visualization for the comparison mode. Color interpretation isn’t a simple process, thus, affecting the time users took to find the correct answer.

Question Q12 can be observed as the one which took users more time to answer, which was expected since it was the set’s most complex question and to be properly answered interaction with several visualizations and filtering process would have to be done. The question minimum registered value was of 18 seconds, which is greater than the majority of the remaining question’s means. Despite values having one of the biggest dispersion values with a standard deviation of 12.64 seconds, the mean is closely symmetric between the box extremes.

Confidence intervals were also calculated based on these 20 test results at a 0.05 significance level and so for further testing we can expect with 95% confidence that the mean for each question will be within the mean $\pm$ confidenceValue.

### 5.3.2 Errors

While interacting with FishEye, user’s errors during the process were registered. Such results are consolidated on Table 5.2.

In total, 16 errors were recorded within the evaluation process. The vast majority of detected errors can be explained by the fact users were not using the correct visualization at first or the lack of use of filter and interactive mechanisms, which can be considered normal and expected since being their first experience within FishEye. Nevertheless, these particular users were able to recover from the errors made and find the right path, on their own, to properly answer the questions.

For the 20 subjects present, 45% (9 users) made at least one error during its evaluation. It is important to note that for each user a maximum of one error per question was recorded, showing users learned from the error made and did not commit any more while answering that particular question. However for six from nine users, new errors were registered on the following questions. The maximum number of errors registered for a particular user was of three errors, for U16.

For the set of twelve questions used, at least one error was recorded for seven questions (58,3%). It was expected that high complexity questions such as Q12, Q10, Q9 and Q8 would be more prone to error, however that wasn’t the case. Few errors were registered for these question, similar to the ones collected for medium complexity questions such as Q5. By observing the total errors (16) for high complexity questions, it can be observed that 63,5% (ten errors) can be related to high complexity questions, showing that a weak relation between errors tendency and question complexity can be established.
Table 5.2: Users number of errors made for each question

<table>
<thead>
<tr>
<th>User</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
<th>Q11</th>
<th>Q12</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U3</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U4</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U6</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U7</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U8</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U10</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U11</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U12</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U13</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U14</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U15</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U16</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>U17</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U18</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U19</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>U20</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

Statistics

<table>
<thead>
<tr>
<th>Min</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0</td>
<td>0</td>
<td>0.05</td>
<td>0.1</td>
<td>0.15</td>
<td>0</td>
<td>0</td>
<td>0.15</td>
<td>0.1</td>
<td>0</td>
<td>0</td>
<td>0.15</td>
</tr>
<tr>
<td>Max</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>0</td>
<td>0</td>
<td>0.22</td>
<td>0.31</td>
<td>0.37</td>
<td>0</td>
<td>0</td>
<td>0.37</td>
<td>0.31</td>
<td>0</td>
<td>0</td>
<td>0.37</td>
</tr>
<tr>
<td>Confidence Interval (95%)</td>
<td>-</td>
<td>-</td>
<td>0.09</td>
<td>0.13</td>
<td>0.16</td>
<td>-</td>
<td>-</td>
<td>0.16</td>
<td>0.13</td>
<td>0.13</td>
<td>-</td>
<td>0.16</td>
</tr>
</tbody>
</table>

5.3.3 System Usability Scale (SUS)

The final step for evaluating FishEye, was regarding the System Usability Scale questionnaire. Users’ questionnaires were consolidated and for each, the SUS score was calculated, following the scoring guidelines provided on the works of J. Brooke [58]. After, the global SUS score for FishEye was computed by applying the mean to all users individual scores. This data is presented as follows in Table 5.3. As it can be observed, the minimal and maximum rating given were 72.5 and 97.5 respectively, on a 100 points scale.

FishEye’s global rating was of 86.125/100 points. To be able to interpret the score’s meaning the works of Miller et al. [8] were consulted. Based on more then 3,500 SUS scores, the authors were able to correlate user scoring with other metrics such as adjective rating, grade rating and acceptability ranges. These correlations can be observed in Fig. 5.2. By correlating FishEye’s global score of 86.125% with the adjacent metrics, we are able to conclude that the achieved score falls into the ranges of what is considered an "Excellent" and "High Acceptable" system with a final grade of "B".
Table 5.3: Users Usability Scale (SUS) item rating

<table>
<thead>
<tr>
<th>User</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
<th>SUS Result (0-100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>77.5</td>
</tr>
<tr>
<td>U2</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>72.5</td>
</tr>
<tr>
<td>U3</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>97.5</td>
</tr>
<tr>
<td>U4</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>80</td>
</tr>
<tr>
<td>U5</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>87.5</td>
</tr>
<tr>
<td>U6</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>85</td>
</tr>
<tr>
<td>U7</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>82.5</td>
</tr>
<tr>
<td>U8</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>85</td>
</tr>
<tr>
<td>U9</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>92.5</td>
</tr>
<tr>
<td>U10</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>85</td>
</tr>
<tr>
<td>U11</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>87.5</td>
</tr>
<tr>
<td>U12</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>87.5</td>
</tr>
<tr>
<td>U13</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>77.5</td>
</tr>
<tr>
<td>U14</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>77.5</td>
</tr>
<tr>
<td>U15</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>90</td>
</tr>
<tr>
<td>U16</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>87.5</td>
</tr>
<tr>
<td>U17</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>95</td>
</tr>
<tr>
<td>U18</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>87.5</td>
</tr>
<tr>
<td>U19</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>97.5</td>
</tr>
<tr>
<td>U20</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>90</td>
</tr>
</tbody>
</table>

Global Score 86.125

Figure 5.2: Grade rankings of SUS scores based on the works of Miller et al. [8]

5.4 Further Statistical Analysis

In order to acquire more knowledge about the obtained results regarding performance metrics (response time and errors), a statistical Kolmogorov–Smirnov normality test was applied. Followed by a Wilcoxon signed rank test to find significant differences between samples, due to evidences against normality found (p < 0.05) in at least one sample for each metric. After, an attempt to find a correlation between performance time and number of errors by computing Pearson coefficients was made.

For both response time and errors, the Kolmogorov–Smirnov test presented some evidence against normality, satisfying the following condition $D_{max} > D_{20.05}$.

Regarding response time the condition was satisfied on question Q1, such that $0.795 > 0.294$. The same question also satisfied the condition regarding errors’ results, such that $0.839 > 0.294$. 

67
Test results showed the data doesn’t present a good fit within the normal distribution, and so suggesting the use of a nonparametric test for further statistical analysis. Having this in consideration, a Wilcoxon signed rank test to find significant differences between samples was applied.

Regarding response time, results can be found on Table 5.4. By observing the table, it can be noted that users took significantly less time to answer the second question than the remaining. The opposite situation can be observed within the twelfth question’s coefficients, showing users took significantly more time to answer Q12 than the remaining. Questions Q1, Q3, Q4, Q7 and Q11 show similar coefficients, being answered, except for Q2, much faster than the remaining. Again, the opposite situation can also be identified within the eighth, ninth and tenth questions, in which users took a much considerably longer to answer these questions than the remaining, excluding Q12.

Finally, for Q5 and Q6 questions, it may be observed that users took much longer to answer these questions when compared to Q1, Q2, Q3, Q4, Q7 and Q11, whereas comparing with Q8, Q9, Q10 led to a shorter time.

Table 5.4: Wilcoxon signed rank test results for response time metric.

<table>
<thead>
<tr>
<th>Question</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
<th>Q11</th>
<th>Q12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>-</td>
<td>2.57</td>
<td>-0.39</td>
<td>-0.66</td>
<td>-3.72</td>
<td>-3.16</td>
<td>-1.6</td>
<td>-3.76</td>
<td>-3.85</td>
<td>-3.72</td>
<td>-1.4</td>
<td>-3.91</td>
</tr>
<tr>
<td>Q2</td>
<td>-2.57</td>
<td>-2.75</td>
<td>-2.75</td>
<td>-3.91</td>
<td>-3.91</td>
<td>-3.78</td>
<td>-3.87</td>
<td>-3.91</td>
<td>-3.91</td>
<td>-3.28</td>
<td>-3.91</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>0.39</td>
<td>2.75</td>
<td>-0.18</td>
<td>0.18</td>
<td>-3.28</td>
<td>-2.34</td>
<td>-0.72</td>
<td>-3.5</td>
<td>-3.46</td>
<td>-3.57</td>
<td>-0.53</td>
<td>-3.91</td>
</tr>
<tr>
<td>Q4</td>
<td>0.66</td>
<td>2.75</td>
<td>-0.18</td>
<td>-3.76</td>
<td>-3.43</td>
<td>-1.17</td>
<td>-3.54</td>
<td>-3.72</td>
<td>-3.8</td>
<td>-1.52</td>
<td>-3.91</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>3.72</td>
<td>3.91</td>
<td>3.28</td>
<td>3.76</td>
<td>-1.37</td>
<td>3.53</td>
<td>-1.13</td>
<td>-1.97</td>
<td>-1.78</td>
<td>3.71</td>
<td>-3.61</td>
<td></td>
</tr>
<tr>
<td>Q6</td>
<td>3.16</td>
<td>3.91</td>
<td>2.34</td>
<td>3.43</td>
<td>-1.37</td>
<td>-2.85</td>
<td>-1.6</td>
<td>-2.25</td>
<td>-2.83</td>
<td>3.48</td>
<td>-3.8</td>
<td></td>
</tr>
<tr>
<td>Q7</td>
<td>1.6</td>
<td>3.78</td>
<td>0.72</td>
<td>1.17</td>
<td>-3.53</td>
<td>-2.85</td>
<td>-3.61</td>
<td>-3.85</td>
<td>-3.91</td>
<td>0.1</td>
<td>-3.91</td>
<td></td>
</tr>
<tr>
<td>Q8</td>
<td>3.76</td>
<td>3.87</td>
<td>3.5</td>
<td>3.54</td>
<td>1.13</td>
<td>1.6</td>
<td>3.61</td>
<td>-0.81</td>
<td>-0.14</td>
<td>3.13</td>
<td>-2.98</td>
<td></td>
</tr>
<tr>
<td>Q9</td>
<td>3.85</td>
<td>3.91</td>
<td>3.46</td>
<td>3.72</td>
<td>1.97</td>
<td>2.25</td>
<td>3.85</td>
<td>0.81</td>
<td>-</td>
<td>0.61</td>
<td>3.69</td>
<td>-2.64</td>
</tr>
<tr>
<td>Q10</td>
<td>3.72</td>
<td>3.91</td>
<td>3.57</td>
<td>3.8</td>
<td>1.78</td>
<td>2.83</td>
<td>3.91</td>
<td>0.14</td>
<td>-0.61</td>
<td>-</td>
<td>3.59</td>
<td>-2.57</td>
</tr>
<tr>
<td>Q11</td>
<td>1.4</td>
<td>3.28</td>
<td>0.53</td>
<td>1.52</td>
<td>-3.71</td>
<td>-3.48</td>
<td>-0.1</td>
<td>-3.13</td>
<td>-3.69</td>
<td>-3.59</td>
<td></td>
<td>-3.91</td>
</tr>
<tr>
<td>Q12</td>
<td>3.91</td>
<td>3.91</td>
<td>3.91</td>
<td>3.91</td>
<td>3.91</td>
<td>3.91</td>
<td>3.8</td>
<td>2.98</td>
<td>2.64</td>
<td>2.57</td>
<td>3.91</td>
<td>-</td>
</tr>
</tbody>
</table>

Regarding errors, the desired test couldn’t be performed, since it requires certain conditions to its realization, such as, to have at least five pairs with different values between sets. Since there only exists a maximum of three different values among the possible set’s combinations, it was impossible to perform such a test.

The final statistical test applied was Pearson’s correlation coefficient test, in order to check if correlations exists between response time and errors. The test results are presented on Table 5.5.

By observing the table seven correlations can be found: Q4 (r = -0.09, p <0.05) presents what is considered a negligible correlation, while Q9 (r = 0.48, p <0.05) and Q10 (r = 0.47, p <0.05) both present a weak correlation, but very close to the minimum limit of what is considered a moderate correlation, in which Q3 (r = 0.63, p <0.05), Q5 (r = 0.58, p <0.05) and Q8 (r = 0.65, p <0.05) are inserted, finally, it can be observed that Q12 (r = 0.73, p <0.05) presents a strong correlation between time response and number of errors.

Having this in consideration, a generalization between question response time and number of errors...
errors correlation can’t be made, given that there is only a strong correlation among questions.

Table 5.5: Pearson’s correlation coefficients between response time and errors

<table>
<thead>
<tr>
<th></th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Q7</th>
<th>Q8</th>
<th>Q9</th>
<th>Q10</th>
<th>Q11</th>
<th>Q12</th>
</tr>
</thead>
<tbody>
<tr>
<td>N.A.</td>
<td>N.A.</td>
<td>0.63</td>
<td>-0.09</td>
<td>0.58</td>
<td>N.A.</td>
<td>N.A.</td>
<td>0.65</td>
<td>0.48</td>
<td>0.47</td>
<td>N.A.</td>
<td>0.73</td>
<td></td>
</tr>
</tbody>
</table>

5.5 Discussion

The performed user evaluation went according to the beforehand prepared guidelines. Users were engaged throughout the entire process and said to enjoy their first experience within FishEye. The SUS questionnaire subjective item’s descriptions can also be used to analyze user’s feelings and thoughts regarding their experience. In fact, by observing the user’s rating distribution among the ten questions (Fig. 5.3), we can perceive that the vast majority of users felt that they would use this system on a frequently basis (Q1), without needing any external help (Q4) or further knowledge (Q10). A quarter of uses were neutral about the system easiness of use while the remaining agreed it was in fact easy of use (Q3). Similar opinions can be found regarding the ease of learning within FishEye (Q7) as also within user’s confidence level while using it (Q9), showing these three metrics (ease of use, ease of learning, and confidence) are strongly correlated. A fully level of agreement among users can be observed regarding the system’s visualizations integration performance (Q5) as also consistency throughout (Q6). All users agreed FishEye doesn’t presents itself as a complex system (Q2).

Having all the described evaluation results into account, we can infer that FishEye was successfully accepted among users, which validates all the work done during FishEye's development process. Furthermore, by being able to answer all purposed questions with none to few errors and a high SUS score, the evaluation assured this dissertation’s main goal: to present marine species’ information in an integrated manner while providing users with new means of learning about fish species and at the same time interacting with such data.
Figure 5.3: User answer distribution for the System Usability Scale (SUS) questionnaire.
Conclusions and Future Work
Currently technology, with tracking and tagging devices, enables easy ocean animal data collection. Consequently, such data is nowadays available in large amounts, often fragmented and unprocessed. As a result, despite its potential, due to the difficulties in analysis, this data is often disregarded. FishEye is a visualization that provides an interactive representation of marine species’ data. It consists of a dashboard with interlinked views that provide the user with complementary views on the existing data, allowing pattern discovery and data interpretation. It also allows the display of information regarding two species at the same time, giving the user the possibility of comparing species across all its existing data domains as also establishing relations and common behaviors about such species. Due to interactive mechanisms such as collapse, resize and drag-and-drop features, FishEye provides each user a personal and more richer experience. To further engage and ease the user into changes FishEye presents several animated transitions across all views.

Following an iterative and incremental methodology, different stages of evaluation took place during development which lead to a more robust and higher quality solution. In the end, usability tests were carried out with 20 users in which twelve questions were asked to be answered by using FishEye. Users were capable of learning about fish species and at the same time interacting with such data, thus being able to answer all questions properly with a minimal degree of error. Afterwards, a SUS questionnaire was made to users which led FishEye to achieve a global score of 86.125/100. Such score is considered “Excellent”.

In order to possibilite users to identify fish species to further learn about them, a species’ recognition module was integrated within FishEye. This includes extracting features about the given input and then classifying it by comparing the extracted features within the FishEye’s existing features database. In order too create FishEye’s feature database, FishEye’s classifier was trained using a total of 4.000 samples. Species samples are added within FishEye as plug-ins, being each plug-in independent from the remaining, allowing the solution to proper scale without further workarounds. To validate the FishEye’s recognition module two species (Clownfish and Surgeonfish) were chosen as a prof of concept. The performed evaluation showed that FishEye can efficiently recognize the trained species within an accuracy of 95%, 100% precision and a real-time response of 1.61 seconds.

Future work includes adding new species to both recognition and visualization processes, being this process independent, since new species can be easily added in the architecture of FishEye, as plug-ins. Furthermore, new idioms should be taken into consideration to be integrated within FishEye’s dashboard. New data attributes should also further researched, thus creating a more complete and meaningful experience as also improving user species’ learning and pattern identification. Ways of decreasing user proneness to error, as also, decreasing time response to properly answer the questions asked during usability tests should also be researched and later carry out further testing to validate such research.
Bibliography


