Computer Vision-based Systems for Environmental Monitoring Applications

by

Tunai Porto Marques
M.Sc., California State University Long Beach, 2016
B.Sc., Federal University of Sergipe, 2013

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical and Computer Engineering

University of Victoria

We acknowledge and respect the lə$\dot{k}^{w}$əŋən peoples on whose traditional territory the university stands and the Songhees, Esquimalt and W̱SÁNEĆ peoples whose historical relationships with the land continue to this day.

Computer Vision-based Systems for Environmental Monitoring Applications

by

Tunai Porto Marques
M.Sc., California State University Long Beach, 2016
B.Sc., Federal University of Sergipe, 2013

Supervisory Committee

Dr. Alexandra Branzan Albu, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Pan Agathoklis, Departmental Member
(Department of Electrical and Computer Engineering)

Dr. Kwang Moo Yi, Out-of-unit Member
(Department of Computer Science, University of British Columbia)

Dr. Lauren McWhinnie, Out-of-unit Member
(School of Energy, Geoscience, Infrastructure and Society, Heriot-Watt University)

ABSTRACT

Environmental Monitoring (EM) refers to a host of activities involving the sampling or sensing of diverse properties from an environment in an effort to monitor, study and overall better understand it. While potentially rich and scientifically valuable, these data often create challenging interpretation tasks because of their volume and complexity. This thesis explores the efficiency of Computer Vision (CV)-based frameworks towards the processing of large amounts of visual EM data.

While considering every potential type of visual EM measurement is not possible, this thesis elects three EM data streams as representatives of diverse monitoring layouts: *visual out-of-water* stream, *visual underwater* stream and *active acoustic underwater* stream. Detailed structure, objectives, challenges, solutions and insights from each of them are presented and used to assess the feasibility of CV within the EM context. This thesis starts by providing an in-depth analysis of the definition and goals of Environmental Monitoring, as well as the Computer Vision systems typically used in conjunction with it.

The document continues by studying the *visual out-of-water* stream via the design of a novel system employing a contrast-guided approach towards the enhancement of low-light underwater images. This enhancement system outperforms multiple state-of-the-art methods, as supported by a group of commonly-employed metrics.

A pair of detection frameworks capable of identifying schools of herring, salmon, hake and swarms of krill are also presented in this document. The inputs used in their development, *echograms*, are visual representations of acoustic backscatter data from echosounder instruments, thus contemplating the *active acoustic underwater* stream. These detectors use different Deep Learning (DL) paradigms to account for the unique challenges presented by each pelagic species. Specifically, the detection of krill and finfish is accomplish with a novel semantic segmentation network (U-MSAA-Net) capable of leveraging local and contextual information from feature maps of multiple scales.

In order to explore the *out-of-water visual* data stream, we examine a large dataset composed by years-worth of images from a coastal region with strong marine vessels traffic, which has been associated with significant anthropogenic footprints upon marine environments. A novel system that involves "traditional" CV and DL is proposed for the identification of such vessels under diverse visual appearances on this monitoring imagery. Thorough experimentation shows that this system is able to efficiently

detect vessels of diverse sizes, shapes, colors and levels of visibility.

The results and reflections presented in this thesis reinforce the hypothesis that CV offers an extremely powerful set of methods for the automatic, accurate, time- and space-efficient interpretation of large amounts of visual EM data, as detailed in the remainder of this work.

Contents

List of Tables

List of Figures

Acronyms

AG Attention Gates. iv, 75

AHE Adaptive Histogram Equalization. iv, 15

AIS Automatic Identification System. iv, 100, 117

ANN Artificial Neural Networks. iv, 51, 52, 69, 70, 102

AUV Autonomous Underwater Vehicles. iv, 8

CCI Contrast Code Image. iv, 25, 34, 35, 38, 40

CLAHE Contrast Limited Adaptive Histogram Equalization. iv, 15

CNN Convolutional Neural Networks. iv, 7, 19, 31, 32, 52, 56, 57, 71, 102, 104, 109, 112, 119, 120

CORAL Coastal and Ocean Resource Analysis Laboratory. iv, 14, 23, 99

CV Computer Vision. iii, iv, x, xv, 1, 9, 10, 13, 21–24, 27, 28, 47, 49, 52, 64, 65, 99, 104, 116, 118–120

DCP Dark Channel Prior. iv, 24, 25, 31–34

DEM Digital Elevation Models. iv, 12

DFO Department of Fisheries and Oceans. iv, 24, 54, 55, 63, 79, 98, 117

DL Deep Learning. iii, iv, x, xv, 16, 20, 21, 23, 27, 47–53, 63, 64, 67–72, 81, 83, 84, 86–89, 91–94, 97, 98, 100, 101, 104, 108, 116, 118, 119

DoF Degrees of Freedom. iv, 12

DPM Deformable Part-based Model. iv, 102

DSM Digital Surface Models. iv, 18

DSMV Detector of Small Marine Vessels. iv, x, 20, 26, 103–105, 108–117

DWT Discrete Wavelet Transforms. iv, 27

EM Environmental Monitoring. iii, iv, viii, xv, 1–5, 8–10, 18, 20–22, 24, 26–29, 47, 48, 63, 65, 98, 99, 103, 104, 117–120

FCN Fully-convolutional Networks. iv, 86

FFK FinFish and Krill. iv, 27, 64, 65, 77, 79, 80, 87, 88, 91–95, 97, 98

FPN Feature Pyramid Networks. iv, 56, 58, 59, 102, 112

GAN Generative Adversarial Networks. iv, 21

GBC Gradient Boost Classifiers. iv, 69, 70

GCP Ground Control Points. iv, 11

GLCM Gray-level Co-occurrence Matrix. iv, 84, 85, 89, 91, 93

GMM Gaussian Mixture Model. iv, x, 26, 99, 104–106, 116, 117

IoU Intersection-over-Union. iv, 58, 89, 91, 95–98, 113, 114, 120

KNN K-Nearest Neighbors. iv, 86, 90

LAI Leaf Area Index. iv, 6

LBP Local Binary Patterns. iv, 84

LIDAR Light Detection and Ranging. iv, 4

LSTM Long Short-Term Memory. iv, 117

ML Machine Learning. iv, x, 50–52, 64, 68–70, 72, 81, 83, 84, 86, 87, 89, 91–94, 97, 98, 118, 119

MLP Multi-layer Perceptrons. iv, 51, 70

MSAA Multi-Scale Additive Attention. iv, 27, 64, 67, 68, 74–79, 96–98, 120

MSE Mean Squared Error. iv, 107, 109

MSS Multispectral Scanner. iv, 6

MSU Mobile Sensor Units. iv, 4

NLP Natural Language Processing. iv, 21, 52, 71

NSERC Natural Sciences and Engineering Research Council of Canada. iv, 63, 98

NVDI Normalized Difference Vegetation Indexes. iv, 6

ONC Ocean Networks Canada. iv, 22, 41, 46

PLHS Pixel-level Herring and Salmon. iv, viii, ix, 50, 53–59, 62, 63, 65, 71, 87, 94–96, 98

PNN Probabilistic Neural Networks. iv, 70

RBN Radial Basis Networks. iv, 70

ReLU Rectified Linear Units. iv

RF Random Forests. iv, 51, 69

RL Reinforcement Learning. iv, 21

RNN Recurrent Neural Networks. iv, 21

ROI Region of Interest. iv, 25, 52, 71, 72, 102

ROV Remotely Operated Vehicles. iv, 8

SGD Stochastic Gradient Descent. iv, 57

SOM Self-organizing Maps. iv, 51, 70

SRKW Southern Resident Killer Whales. iv, 100, 101, 111

SVM Support Vector Machines. iv, 25, 51, 69, 70, 86, 90, 91, 93, 119

TM Thematic Mapper. iv, 6

TOA Top of Atmosphere. iv, 13

UAV Unmanned Aerial Vehicles. iv, 4–7, 101

WSN Wireless Sensor Network. iv, 3

THESIS STATEMENT

Long-term Environmental Monitoring (EM) represents a critical tool towards the understanding of diverse natural ecosystems. The potentially immense amounts of visual monitoring data collected by EM initiatives, often unfeasible to be manually processed, can be interpreted using a host of Computer Vision (CV)- and Deep Learning (DL)-based techniques in an effective and timely manner. There is a flagrant need for application-specific solutions that address the particular challenges associated with the analysis of diverse streams of visual EM data. This thesis details four case studies showing that the use of CV and DL within an application-specific paradigm is feasible for interpreting EM data.

Chapter 1

Introduction

Environmental Monitoring (EM) initiatives are capable of producing rich and temporally representative data that can be processed into information, and eventually knowledge. The expansive nature of these measurements with variable timespans hinders, or often completely prevents, a meaningful manual interpretation of them (see sub-section 1.2). In an attempt to explore and reflect about efficient ways to autonomously interpret EM visual data, the present work attempts to answer the following research question:

> *Can Computer Vision techniques be broadly employed to efficiently and accurately process years-worth of Environmental Monitoring visual data, given their unique natures, challenges, and scientific considerations?*

Before defining the thesis structure (Figure 1.6) and objectives, this Chapter motivates this research endeavour by defining Environmental Monitoring and discussing its importance and applications in sub-section 1.1. A thorough discussion on the different types of Optical Systems used in EM, data-related interpretation challenges, as well as common Computer Vision (CV) methods typically employed in this context ensues, respectively, in sub-sections 1.2 and 1.3. Finally, considering the context offered in the rest of the Chapter, the specific Objectives and Contributions of this work are outlined and detailed in sub-section 1.4.

1.1 Environmental Monitoring: Definition, Importance and Examples

Environmental Monitoring involves the sensing or sampling of diverse properties (typically employing three major scientific disciplines: physics, biology and chemistry) in one or multiple sites over extended time periods, from which one can derive knowledge. The overall goal of such efforts is to better understand earth's cycles (*e.g.*, water, nitrogen) and their interactions with biological organisms, geological components, the atmosphere, among others [1]. The data gathered in EM can support the creation of well-informed public policies and protection of natural resources, motivate scientific programs, as well as lead to a better understanding of the impacts of human activities on different environments [2]. Environmental Monitoring is critically important because policies proposed in environmental sciences often require that the target environments have been previously monitored and well-understood [1].

The scientific value of efficient and acurate EM systems is rooted in the fact that their observations and measurements provide reliable data upon which actionable knowledge can be generated. This knowledge should provide a rich understanding of the environment, thus enabling informed decisions [1]. This data-based understanding can be applied in a plethora of ways that range vastly in scope: it might serve, for example, as the basis for the creation and validation of environmental models (by determining relevant parameters) that can predict the changes expected in the climate as a result of rising greenhouse gas emissions [3]. It can also be used in smaller-scale, more focused studies, such as the analysis of the impacts on fish from human-generated fluctuating thermal conditions in lakes [4].

Gary *et al.* [5] highlights some nation-wide examples where environmental monitoring programs, while providing crucial insights and actionable data, involved only negligible fractions of the implementation costs of U.S. environmental policies:

1. Clean Water Act (CWA): Created in 1972, this U.S. federal law establishes a framework for regulating the discharging of pollutants in lakes, rivers and coastal waters. Estimates place the national cost of complying to this law between $14.1 billion [6] and $93.1 billion [7] annually. On the other hand, monitoring the water to observe the effectiveness of the water-cleaning measures put in place and eventually change them costs only an approximate 0.4-2.1% of that amount. This example illustrates how well-designed monitoring programs could

guide stakeholders to the implementation of optimal (and lower-cost) activities towards the compliance with the CWA.

2. Clean Air Act: A federal law from the U.S. was created in 1963 to enforce the maximum acceptable levels of air pollution. An amendment from 1990 called the Acid Rain Program (ARP) was added to the Clear Air Act to address the specific problem of acidifying deposits of sulfur and nitrogen. By 2007, a reduction of seven millions tons in the sulfur dioxide levels and a reduction of over three million tons of the nitrogen oxide levels had been attributed to the ARP program (with respect to 1980 and 1990 references, respectively) [5]. The annual estimated cost for the compliance with this program is $3 billion, while its projected benefits surpass $120 billion [8]. The cost of air monitoring related to the ARP program is significantly lower, representing only 0.4% of the ARP implementation cost, and less than 0.01% of its predicted benefits [5].

Environmental Monitoring also plays a critical role in the evaluation of environmental policies. In a 2005 study, Bernhardt *et al.* [9] exemplified a situation when the lack of environmental monitoring was particularly damaging: an expenditure of $13-14 billion in lake restoration projects in the U.S. (since 1990) often did not have monitoring data before and after the restoration efforts, ultimately preventing an effective analysis of their effectiveness.

Sensors in EM. Environmental Monitoring can be performed using a multitude of sensors, each focusing on a particular type of data. Fixed sensors that gather *in-situ* data have been used to monitor physical phenomena such as temperature, light, sound, pressure, as well as chemical characteristics like air [8, 10] and water [7, 11, 12, 13] pollution levels. As an example, Lutakamale and Kaijage [14] created a Wireless Sensor Network (WSN) that measures temperature, humidity and smoke, aiming for the early detection of wildfire in Tanzania. As typically done in WSNs, the measurements are logged into each individual sensor (or *node*), and an internet-based infrastructure composed of micro-controller, communication protocol and messaging platform is used to transmit the collected data. Two successful examples of WSNs designed to monitor water quality are the *SmartCoast* [13], which measures parameters such as dissolved oxygen, conductivity, pH level and turbidity, as well as the system proposed by Jiang *et al.* [12], more suitable for temperature and pH level measurement in large-scale environments.

The main drawback of fixed sensors lies in their inability to monitor increasingly

vast areas. Solutions that use mobile robots carrying Mobile Sensor Units (MSU), allowing for the coverage of bigger sites, were presented in [15, 16]. Trincavelli *et al.* [15] proposed a mobile unit that could detect gas distribution (*i.e.*, hydrogen, carbon monoxide, methane, ammonia) in indoors and outdoors environments, while Weibring *et al.* [16] used a mobile Light Detection and Ranging (LIDAR) platform to identify pollutant emissions and do environment imaging. Hybrid approaches have also been proposed, for example the system of Hes *et al.* [17], which tackles the static location limitation by using bees as mobile sensors in what they named biological and robotic sensor networks. Each individual insect (*biological sensor)* contributes with some information that is collected by monitoring the overall health of their colony. The goal in [17] is to use chemical information from the insect's colonies to determine possible contamination from the bees' food sources. Mobile robots equipped with MSUs perform on-site analysis to confirm or refute the contamination suspicion.

1.2 The usage of Optical Systems in Environmental Monitoring

Optical systems are sensors that interact with light in different wavelengths (*e.g.*, visible range, infrared, ultraviolet) to create images used for monitoring areas that can vary drastically in size, as detailed in Table 1.1. This table clusters different applications that employed optical systems in EM based on their overall level of coverage area (from wide to narrow), as well as the type of environment they are based on: *satellite imagery* can be used to monitor enormous geographical areas and study large-scale land cover changes [18]; *aerial imagery*, commonly obtained with the use of Unmanned Aerial Vehicles (UAV), allows for the real-time study of vegetation, wildlife, as well as precise 2D and 3D mapping of sites [19]; *coastal imagery*, which usually gathers data from a particular segment of a coast, helps understand the geographical changes over time on these environments [20]; *in-situ imagery* provides a greater level of granularity (at the expense of coverage area), allowing researchers to study specific individuals (or groups of), ranging from small insects [21], going through medium-sized mammals [22], up until big mammals [23]; *underwater imagery* (not necessarily related to the proportion of covered area) assists in the study of the ecology [24] and geography [25] of underwater sites.

Optical systems provide great amounts of data with rich scientific potential. For

Table 1.1: Clustering based on coverage area and deployment site of relevant works in Environmental Monitoring that use data from optical systems.

Type of optical data	Common goals	References
Satellite imagery	Monitor deforestation [26], study land cover changes [27, 18], perform 2D/3D mapping [28], sense biodiversity [29], enforce public policies [30].	[27, 18, 31, 32, 33, 34, 35, 36, 29, 37, 38, 39, 40, 26, 30, 28]
Aerial imagery (*e.g.*, UAV-based)	Photogrammetry [41], 3D modeling [19, 42], post-disaster analysis [43, 44, 45], vegetation monitoring [46, 42], large-scale wildlife research [47, 48].	[47, 19, 41, 49, 37, 50, 46, 51, 42, 48, 52, 53, 43, 44, 45]
Coastal imagery	Assess the changes in coastal regions based on natural and human-driven events [54, 41, 55], analyze the frequency of objects of interest (*e.g.*, marine vessels [56] and litter [57]).	[58, 54, 41, 59, 20, 60, 61, 56, 55, 57]
In-situ imagery	Study specific species and individuals [62, 63, 21], count specimen [23], model animal behavior [22], small- and medium-scale wildlife research [22, 64, 65].	[62, 63, 22, 64, 21, 66, 23, 65, 66]
Underwater imagery	Study the ecology and geology of underwater sites [67, 68, 25], enhance visual data [69, 70, 71, 72], track and count specimen [73, 74], discover new species [75], perform acoustic-based monitoring [76, 77, 78].	[73, 67, 68, 25, 70, 79, 80, 24, 81, 76, 77, 74, 69, 78, 71, 72, 82, 83, 84, 85, 86, 87]

example, images from satellites such as the Landsat [88] can often be used to perform analyses that might have not been initially envisioned when the optical systems were deployed: Vittek *et al.* [32] used data from the Multispectral Scanner (MSS; first satellite sensor for Earth observation, deployed in 1972) and its predecessor, the Thematic Mapper (TM), to study the land cover change over West Africa between 1975 and 1990. Some of the images acquired with Landsat's MSS/TM optical systems were already more than forty years old when used in multiple other works (*e.g.*, [34, 35, 36]), attesting to the multi-purpose nature of such data. Another example of the rich potential of data acquired with optical systems is the Snapshot Serengeti dataset [89], composed by 6.7 million labeled images of 55 different animal species. This dataset was used to carry citizen science projects [90], study the ecology of certain environments [91] as well as train robust, deep learning-based image classifiers [65, 23]. Common goals, uses and applications of each of the five types of optical data detailed in Table 1.1 are discussed below:

1. **Satellite imagery.** Optical systems deployed in satellites (e.g., Multispectral Scanner and Thematic Mapper) have the ability to create large-scale images ranging from an area coverage of 30 m^2 per pixel up until images of entire continents (Landsat satellite) in different wavelength ranges (*e.g.*, visible [32], infrared [31]). Chen and Cihlar [35] combined satellite data and *in-situ* experiments by obtaining ground-based measurements of Leaf Area Index (LAI) from two Canadian Boreal Conifer Forests, and correlating that data with Normalized Difference Vegetation Indexes (NVDI) from Landst's TM satellite imagery. Their *in-situ* measurements were performed using two other optical instruments: the Plant Canopy Analyzer (LAI-2000) and the Tracing Radiation and Architecture of Canopies (TRAC). The authors concluded that the Landsat TM imagery can, in fact, be used to accurately infer the LAI of boreal forests (given certain weather conditions).

2. **Aerial imagery.** These data can be gathered using aircraft of different sizes: airplanes, helicopters, aerostats, or most commonly UAVs. The main difference between aerial and satellite imagery in that the former is cheaper and more easily accessible (*i.e.*, does not have satellite infra-structure costs, data licensing fees), allowing for custom deployments focused on specific areas and projects. Siebert and Teizer [52] created a low-cost (approximately $8,000 in 2014) autonomous UAV-based platform for photogrammetric surveying, which is the process of ob-

taining information about physical structures based on images. Their proposed UAV system follows a predetermined flight path, captures geo-referenced images at given intervals, then provides the data to a novel software that creates a 3D map of the monitored environment. The authors evaluated the system in construction work sites, and stated that the mapping of rapidly-changing environments (*e.g.*, sand dunes) could also benefit from their technology.

3. **Coastal imagery**. Although the nearshore areas represent only thin regions that delimit continents, their relative societal value is extremely high. There is a significant recreational, defense, ecological and overall economical interest (*e.g.*, over $3 trillion of U.S. infra-structure is installed there [54]) in such environments. Marine geology, which studies the transport of sand by currents, sand bar development, beach erosion, and overall beach morphology, plays an important role in the monitoring of nearshore areas. This field of study can greatly benefit from coastal imagery, mainly because nearshore processes present rich visual signatures [54]. For example, the water depth influences the appearance of waves, so coastal imagery can be used for bathymetry estimation (*i.e.*, study of underwater depth). The monitoring of marine vessels and species based on cameras placed at nearshore areas also represents a typical use of coastal imagery [56].

4. ***In-situ* imagery**. In this layout of optical systems one can analyze a very specific and well-delimited area. The infra-structure is typically composed by cameras (either mobile or statically located) deployed in the monitored sites. A common strategy for the gathering of *in-situ* imagery are the camera-traps, motion sensor-enabled optical systems that record scenes only when animal specimen are present. Visual data describing behavior and location of animals in the wild can play an important role in ecosystem's understanding and conservation. Ecology, wildlife and conservation biology, zoology, among others, are some examples of study fields that can benefit from *in-situ* imagery [23]. A common challenge faced by researchers is the manual interpretation of the often overwhelming amounts of data captured using these systems. Norouzzadeh *et al.* [23] used 3.2 million wildlife images from the Snapshot Serengeti dataset [89] to train a system capable of autonomously detecting, counting, and providing limited additional information on the animals present on each sample. The proposed system uses Convolutional Neural Networks (CNN) to

extract meaningful visual features of each image, which then serve as input for Fully-Connected layers, responsible for the final classification score of each candidate sub-region of the image.

5. **Underwater imagery**. Although sometimes overshadowed by the amount of effort put into the analysis of terrestrial sites, the study and preservation of the oceans (which occupy almost 71% of Earth's surface) is just as important. Marine sites host extremely rich ecosystems and meaningful geological structures that must be understood and preserved. Roff and Zacharias [92] exemplify the ocean's importance by highlighting its ability to control both its own and the land's climate, playing a vital role in efforts against global warming. The authors enumerate a series of factors that attest to the ocean's multifaceted value: *intrinsic* value, which pertains to the biological value of the species living underwater; *anthropocentric* value, involving renewable and non-renewable resources, as well as goods and services related to the oceans; *ethical* value, which states that for humans as part of the nature, we should be fighting for its conservation. Underwater optical systems provide the time series of data necessary to monitor and study these environments. Friedman *et al.* [80] used a combination of Autonomous Underwater Vehicles (AUV), Remotely Operated Vehicles (ROV) and diver-held optical systems to capture geo-referenced underwater images. These images were later used to perform bathymetric reconstructions of the ocean floor, and finally to calculate measures of rugosity and slope of the monitored sites. The efficacy of this image-driven method was demonstrated by comparing its results with *in-situ* measurements done with other instruments in large (*i.e.*, up to 3,750 m^2) areas. The image-driven reconstruction of the underwater site not only performs as well as its *in-situ* counterpart, but it also requires less infra-structure and creates less disturbance in the monitored areas.

Challenges on the interpretation of visual EM data. In order to harvest all the potential information that images can offer to Environmental Monitoring, a manual, expert-led analysis of the data is often necessary. For example, acoustic backscatter data visualized as an image (*i.e.*, echogram) might provide rich information about the composition, behaviour and abundance of marine species [76, 77, 82, 83, 84, 85, 86, 87]. The interpretation of these non-natural-looking visual data involves biological and acoustic knowledge that is typically held only by experts. As a result, manually processing these years-worth of underwater acous-

tic data involves time-consuming and error-prone processes that might also lead to inter-expert disagreements [76, 77] (see sub-section 3.1 for details).

Other streams of visual EM data that involve time-consuming manual processing consist of the enormous amounts of imagery coming from decades of satellite monitoring [18, 34, 32], multiple years of camera-trap's output [89, 66, 22], billions of instances of individual fish in large marine ecosystem's video datasets [81], among others.

These processing challenges can sometimes be mitigated with the use of public science or crowdsourcing services [1], but such alternatives might not be feasible when time and funds are constrained, expert knowledge is necessary or sensitive data is used. It therefore became clear to environmental researchers that an elegant and systematic way to interact with these EM visual data (in particular if the analysis done by experts could be mimicked) is of vital importance. Computer Vision presents itself as a powerful field where techniques for the autonomous interpretation of imagery can be leveraged by Environmental Monitoring, as detailed in the following sub-section.

1.3 Computer Vision and Environmental Monitoring

La Salle [93] describes the need for a fast and meaningful analysis of visual data related to environmental monitoring to create a "digital biodiversity knowledge bank" that would help mitigate major challenges for the next three decades: food security, emerging diseases, managing scarcer natural resources. The author argues that current and future technologies on Computer Vision provide an scalable approach that can efficiently detect animal phenotype, determine biodiversity and natural resources levels, among others. The three main considerations to be made about the consolidation of the collection and use of visual data in environmental studies are: increase by orders of magnitude the rate at which the images are acquired; establish frameworks for the storage, distribution and discovery of such data; provide tools that assist in the extraction of meaningful information from these digital libraries.

Environmental Monitoring poses challenges that can both benefit from the techniques developed in Computer Vision, as well as stimulate research in this field. Differently from other monitoring sensors that provide more focused measurements

[1] https://www.mturk.com/

(*e.g.*, a single number representing the temperature of an entire room obtained by a thermometer), visual data are complex, rich outputs that provide multiple measurements in each of their *pixels*. The intensity of each pixel represents a physical aspect that varies based on the radiation (light) wavelength range being measured: in the *visible* range, pixels represent the amount of visible light that was reflected by the surface of objects; pixels from the *infrared* range can be used to sense the amount of heat being generated by each component of a scene. Some optical systems can also combine the result of radiation measurements obtained in multiple wavelengths (hyperspectral imaging) to extract useful information from visual data [68, 25, 79, 39].

Imagery collected from environmental monitoring is inherently challenging for Computer Vision algorithms because the real world offers objects of interest with variable appearance (*i.e.*, different poses, scales, illumination, occlusion). The goal then becomes to find their *invariant features* (those that can be noticed even under different conditions), so that information relevant to different tasks (*e.g.*, classification, segmentation, detection, reconstruction) can be extracted [94]. Since more than 50% of the human cortex is devoted to processing visual information [95], these invariant features, as well as the overall meaning of a composition of individual pixels (or real scenes) is effortlessly understood by humans, making them the ideal candidates for the interpretation of visual data. However, specific applications that require prior contextual knowledge, big amounts of data and complex operations done with images warrant the development of methods for the automated processing and interpretation of images (often referred to as *image processing* and *image understanding*, respectively [94]), which are two critical pillars of Computer Vision.

1.3.1 Computer Vision Techniques Commonly Used in Environmental Studies

Multiple Computer Vision methods are used to pre-process and interpret visual data acquired by environmental monitoring systems. Although it is not feasible to discuss the entirety of those methods, this sub-section aims to briefly introduce some relevant Computer Vision techniques recurrently used in the analysis of visual data obtained by Environmental Monitoring programs.

Orthorectification. Optical distortions caused by the geometry of the sensor used to capture visual data and the geography of the monitored terrain pose important problems for the use of aerial and satellite imagery. In order to perform a consistent

analysis of such images or the accurate overlaying of photos and maps, one must certify that all the pixels of an image represent an orthogonal projection of all points on the ground with respect to a reference surface [96]. Images that comply with that requirement are called orthoimages or orthophotos. Different models can be used to achieve this goal, and might consider some of the possible distortion sources: sensor attitude (*i.e.*, roll, pitch, and yaw angles) and altitude, earth shape and rotation, systematic errors of the optical system.

Polynomial rectification is considered to be the simplest way to rectify distorted images. Novak [96] describes the transformation between original and rectified image using Equation (1.1).

$$\begin{aligned} x &= x'^T \mathbf{A} y' = f_x(x', y') \\ y &= x'^T \mathbf{B} y' = f_y(x', y') \end{aligned} \tag{1.1}$$

where x, y represent the pixel coordinates of the original image, x', y' are the coordinates of the rectified (reference) image, and $\mathbf{A}, \mathbf{B}$ are coefficient matrices of the polynomials. The x'^T term is given by $x'^T = (1, x', x'^2, x'^3, ...)$, depending on the order of the polynomial chosen (y'^T is calculated in a similar fashion). This method maps reference points of an orthoimage (x', y' coordinates) into its equivalent location in the input image. These references are given by Ground Control Points (GCP), which represent geographical surface locations where the map coordinates (i.e., degrees of latitude and longitude) are known. The polynomial coefficients of the $\mathbf{A}, \mathbf{B}$ matrices are calculated based on the GCPs.

Projective transformation, typically employed with images of flatter surfaces, uses the relationship between two planes (given by eight parameters) to compensate for optical distortions. Figure 1.1 illustrates two planes and eight control points. Each pair of control points provides four parameters.

For aerial/satellite imagery, a visible object placed in the monitored surface with known map coordinates can provide a pair of control points (*e.g.*, 1 and 1′ of Figure 1.1), where the coordinates of these points offer a total of four parameters. Homography transformations are used to project the coordinates of a same point in a given plane p_a (x_{p_a},y_{p_a}) to a second plane p_b (x_{p_b},y_{p_b}) following Equation 1.2:

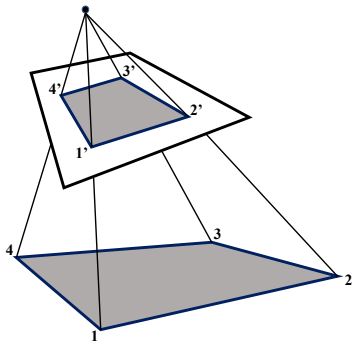


Figure 1.1: Eight parameters are sufficient to determine the relationship between two planes. Each control point (numbers in the image) provides two parameters. These parameters can be used to determine a homography matrix capable of rectifying the optical distortions of an image plane if a second plane is used as reference.

$$\begin{bmatrix} x_{p_b} \\ y_{p_b} \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_{p_a} \\ y_{p_a} \\ 1 \end{bmatrix} \tag{1.2}$$

where the $\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$ term (homography matrix) is typically normalized with $h_{33} = 1$, resulting in 8 Degrees of Freedom (DoF). The calculation of these eight parameters $(h_{11}, ..., h_{32})$ can be viewed as a minimization problem (solvable using, for example, Constrained Least Squares as described in [97]) whose solution only requires eight *known* variables. Therefore, two pairs of control points (eight coordinates) are enough to rectify a distorted input image using Equation (1.2) [96]. Additional reference points in planes p_a and p_b can be used to refine the calculation of parameters from the homography matrix. Parameters related to the exterior and interior orientations of the two planes are not necessary, as they are implicitly expressed through these eight parameters [96]. Similarly to what is done in the polynomial rectification, this method can map points in the reference plane (x_{p_a}, y_{p_a} coordinates) to the input image (x_{p_b}, y_{p_b} coordinates).

Another popular geometrical correction method is the *differential rectification*, where each small region of an image is rectified at a time by applying scaling, shifts and rotations (and comparing those with sub-regions of a reference). Specifically for digital images, each pixel (instead of sub-region) is transferred from the input to the output at a time. Digital Elevation Models (DEM), which describe the physical elevation of each point based on a reference plane, are then used to correct relief displacements (*i.e.*, discrepancy between where an object appears to be versus where

it physically is). Other rectification methods rely on physical models of the optical systems used, and are hardware-specific.

Radiometric corrections. This category of correction techniques deals with changes (or "transformations" [98]) in the intensity levels of individual pixels with the goal of compensating for weather conditions, low-contrast, color inconsistencies, atmospheric interference, among others. For example, in satellite and aerial images the radiometric measurements represented by the intensity of individual pixels may include not only the radiation reflected by the surface (desired), but also that reflected and scattered from the illumination source by atmospheric elements. Top of Atmosphere (TOA) reflectance indicates the total amount of radiation observed by the optical sensor, which includes the undesired radiation coming from sources other than the surface's reflection or emission (in the case of thermal imaging). The raw data obtained from satellite optical systems (*e.g.*, Landsat [18]) is usually accompanied by metadata that can be used to convert it into TOA reflectance ($\rho\lambda$) according to Equation (1.3) [99]:

$$\rho\lambda = \frac{M_\rho Q_{cal} + A_\rho}{cos(\theta_{SZ})} \tag{1.3}$$

where Q_{cal} is the raw pixel intensity value, and the following are metadata-provided parameters: local solar zenith angle θ_{SZ}, additive and multiplicative rescaling factors A_ρ and M_ρ, respectively. The surface reflectance values are obtained by applying atmospheric correction models to TOA reflectance images. These models require the knowledge of physical atmospheric phenomena such as the amount of water vapor and distribution of aerosols, which are often difficult to measure. Simplifications like the Dark Object Subtraction method [100] consider one single pixel value as a representative of the atmospheric influence (*i.e.*, reflection and scattering of source radiance), and then perform an uniform pixel intensity subtraction throughout the entire image.

In-situ imagery also suffers from the effects of scattering and reflection from the light source's radiance, both caused by small physical particles (*e.g.*, dust or water droplets) suspended in the air. Given the abundance and relative ease in obtaining this category of imagery, it comes as no surprise that there exists an entire sub-field of Computer Vision—*image dehazing*—dedicated to the mitigation of haze effects. Chapter 2 details a novel framework proposed for performing single-image dehazing, and explores a case study where low-lighting underwater images are enhanced [101,

70, 69]. Figure 1.2 shows the effect of utilizing the single-image dehazing framework proposed. Note that the marine vessel is barely visible in the original, hazy image (Figure 1.2a). After the dehazing process, the vessel becomes visible and presents a higher contrast, a common characteristic of dehazed images (Figure 1.2b). This process might reveal important information hidden in hazy visual data, ultimately boosting the performance of both manual or autonomous environmental analyses.

(a) (b)

Figure 1.2: Dehazing example in a still image captured by UVic's Coastal and Ocean Resource Analysis Laboratory (CORAL) in Saturna Island, BC, overlooking the Boundary Pass strait. The visibility of the original image (a) is greatly reduced by haze. In (b), the proposed image enhancement framework (detailed in Chapter 2) creates a dehazed output where the visibility of a marine vessel is greatly increased (yellow dotted bounding box).

Contrast enhancement is another common radiometric correction routine. Images that have an unbalanced distribution in the number of pixel intensities r_k (kth intensity level in the interval [0,L], L is typically 255) will show a low contrast, as that on the left of Figure 1.3. A histogram h is a discrete function that helps visualize this intensity distribution using $h(r_k) = n_k$, where n_k is the number of pixels in the image that have intensity level r_k. The cumulative histogram H ($H(k) = \sum_{j=0}^{k} n_k$) of an image reflects the sum of pixels with a given intensity (*e.g.*, $H(100)$ specifies the sum of pixels whose intensities are 100 or lower).

The histogram of images with low contrast helps visualize the unbalanced distribution of n_k for the different pixel intensities r_k (*e.g.*, peak around $r_k = 200$ in Figure 1.3 left). A more homogeneous distribution of n_k boosts the contrast of an image and helps in its visualization and interpretation, as illustrated in the second and third columns of Figure 1.3. A common and simple technique used to achieve better intensity distributions is called *linear histogram equalization*, as described by Gonzalez *et al.* [98]:

$$S_k = \sum_{j=0}^{k} \frac{n_j}{n} \tag{1.4}$$

where S_k is the intensity of a pixel in the output image whose original value was r_k, k ranges in the integer interval $[0, L-1]$, n is the total number of image's pixels and n_j is the number of pixels with intensity j in the input. The center column of Figure 1.3 illustrates the results of applying this histogram equalization technique.

The haze present in the original image (Figure 1.3 left) turns it brighter, resulting in a histogram with intensity distributions more concentrated in higher values. After the linear equalization, the histogram presented in the middle column of Figure 1.3 reveals that the intensities are quasi-homogeneously distributed throughout the $[0, 255]$ range. The visual result of this operation is an image that better highlights objects, patterns, edges and corners. The central column of Figure 1.3 also shows that the cumulative histogram of the histogram-equalized output follows a linear progression, which does not always produce natural-looking images.

In order to mitigate this effect, *histogram matching* approaches change the intensities of the input image in an attempt to approximate its cumulative histogram to that of a reference image. Another popular contrast-enhancing method is called Contrast Limited Adaptive Histogram Equalization (CLAHE) [102], where only the vicinity of a pixel is considered when performing the enhancement (this method differs from the Adaptive Histogram Equalization (AHE) [103] by limiting the amount of amplification applied to the intensities). The rightmost column of Figure 1.3 illustrates the output of CLAHE. Note that local details of the image such as the rocks, water patterns and waves' foam in the seashore are greatly highlighted in this output.

Image segmentation. Images used in environmental monitoring often contain visual elements that do not contribute towards the interpretation of the data. Dividing the image into distinct segments can help in determining, for example, what should be considered as *foreground* or *background* in a scene. Wulder *et al.* [104] determined the time since the last harvest in forest regions by calculating polygons inside which pixels had similar properties, thus segmenting them from bigger satellite images. In underwater settings, often times the medium (water) is irrelevant to the study of diverse animal phenotype or abundance. Padmavathi *et al.* [105] proposed a method to segment underwater objects from the medium, helping focus the analysis of such

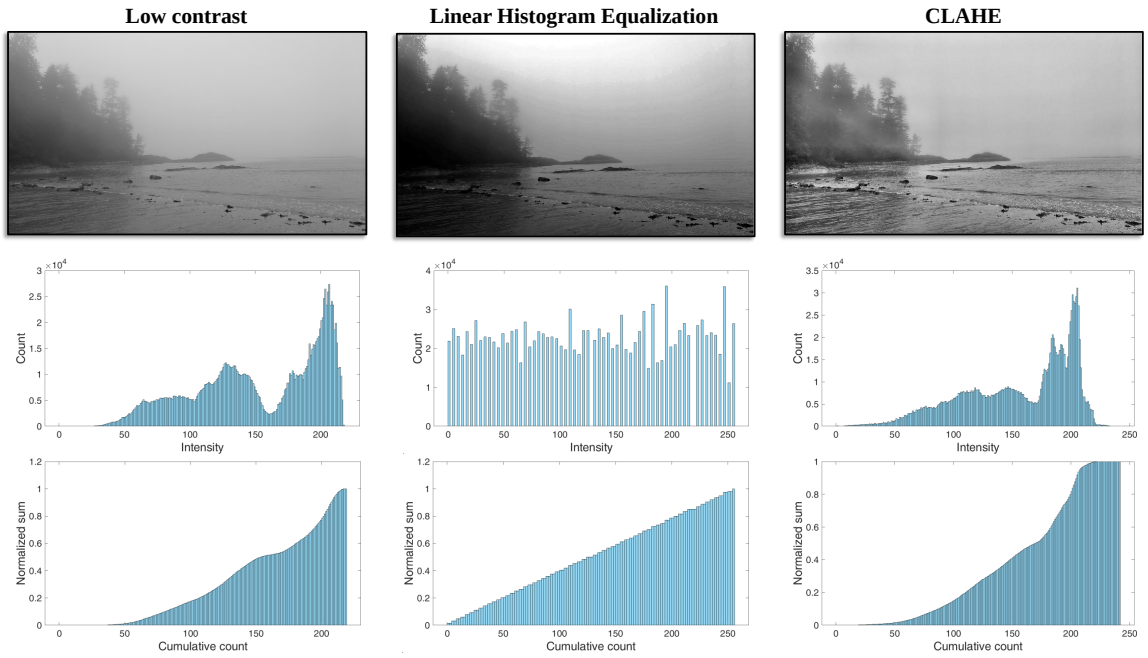


Figure 1.3: Effects of contrast enhancement. From top to bottom in each column: images, their regular and cumulative histograms. **Left column**: grayscale, low contrast image from a beach scenery. Its histogram illustrates the low contrast with a disproportional concentration of pixels in higher intensities levels. **Central column**: output of a linear histogram equalization. Note that the histogram of this image is more homogeneously distributed, indicating a higher contrast. **Right column**: output of the CLAHE [102] method. Details of the image are more easily visualized because of the local awareness of this contrast-enhancing approach.

environments (in a system sharing a similar goal, Akkaynak and Treibitz [106] used RGBD images to infer backscatter coefficients and remove water-related colorcast from images).

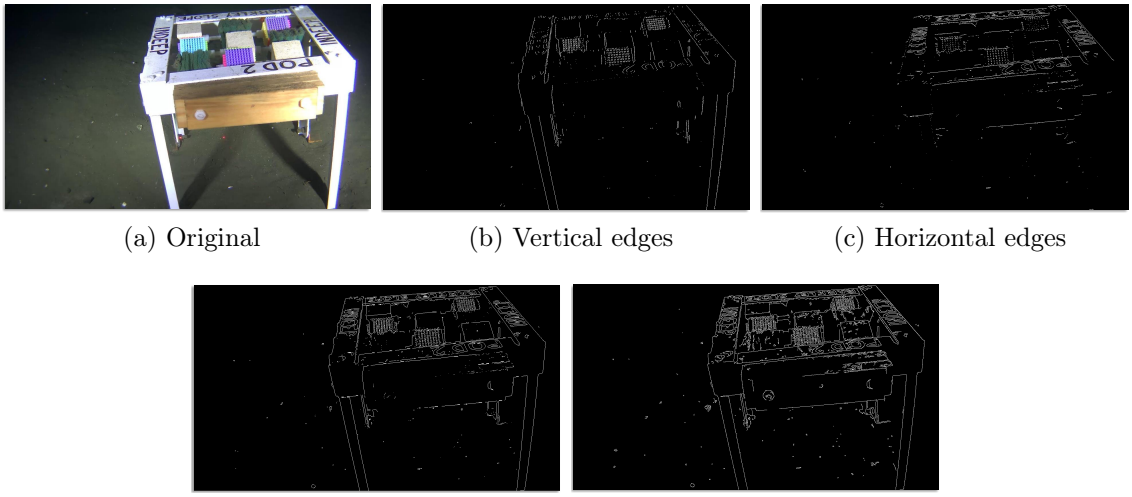
Simple segmentation techniques are typically based on one of two properties of pixel's intensities: discontinuity or similarity [98]. In the former, abrupt changes in pixel intensity represent points, lines, corners or edges that can be used to create boundaries around regions to be segmented. *Edge detection* is a common task in image processing because it can be used for a number of applications such as object detection, thresholding, segmentation, motion tracking, image registration, among others (note that DL-based methods have also been recently used to accomplish such goals). Edges are detected with the use of derivatives that indicate regions in the image where discontinuities are meaningful (*i.e.*, their *magnitude* is bigger than a threshold).

In images, approximations of first- and second-order derivatives can be obtained by calculating differences in pixel neighborhoods of certain sizes (using edge detection kernels). Sliding these masks through an image and then thresholding the results generates binary images that highlight exclusively the image's edges. The *Sobel operator* is a kernel formed by the convolution of an approximation of a first-order derivative, $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$ (to identify edges), and a Gaussian kernel, $\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ (to mitigate the effects of noisy inputs). Sobel operators are thus commonly used to identify vertical and horizontal edges using, respectively, the following kernels: $\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$, $\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$. More robust pipelines, such as Canny's edge detector [107], smooths the image using a Gaussian filter, calculates edges using a given kernel (*e.g.*, Sobel, Prewitt [108], Roberts [109]), performs non-maximum suppression based on the magnitude and orientation of gradients, and finally connects "strong" and "weak" edges based on arbitrary thresholds.

Figure 1.4 illustrates the use of different edge detection approaches. Figures 1.4b and 1.4c show, respectively, the vertical and horizontal edges identified using Sobel operators. Figure 1.4d presents a combination of these two groups of edges, while Figure 1.4e highlights the fact that the Canny edge detection method [107] is able to create more cohesive sets of edges by taking into account the magnitude and orientation of the gradients. The edges and overall shapes represented in the binary images of Figures 1.4d, 1.4e can be used to segment relevant foreground data (*e.g.*, the scientific instrument) from background data.

Segmentation efforts based on similarity, or *region*-based, try to compose regions of pixels that satisfy a given property (*e.g.*, "*all pixels inside region* R_1 *have the same intensity level*"). The image is sub-divided in different regions whose individual pixels satisfy this condition, but that are disjoint with respect to other sub-regions. Gonzales *et al.* [98] enumerates the most common region-based segmentation approaches: *region growing*, which groups similar pixels into sub-regions starting from given seed locations; *region splitting and merging*, where an image is divided into a set of arbitrary, disjoint regions, and then progressively re-grouped into concise regions, also based in a similarity criteria; *watershed transform-based segmentation*, which interprets the intensities of grayscale images as heights, and then looks to find a line (watershed ridge) that divides two depressions (catchment basins) in this topographical map, thus dividing the image into distinct sub-regions.

Features for image description. A typical task that follows image segmentation is to describe a sub-region, or object of interest, using a set of predefined and



(a) Original (b) Vertical edges (c) Horizontal edges

(d) Horizontal and vertical (e) Canny edge detector [107]

Figure 1.4: Edge detection using an approximation of first-order derivatives and the Canny edge detection [107]. (a) Original image from the OceanDark dataset [70]. (b) Vertical edges obtained using the Sobel operator. (c) Horizontal edges obtained using the Sobel operator. (d) Combination of horizontal and vertical edges. (e) Output of the Canny edge detection method.

distinguishing keypoints (henceforth referred to as *features*—note that this term often represents the combination of keypoints and their descriptors *e.g.*, [110]). Examples of Environmental Monitoring applications where visual features can be used are:

1. Object classification. By extracting previously hand-crafted features from candidate regions, one can identify relevant object classes (*e.g.*, fauna, flora, geological structures), as done in [66] for the recognition of animal species and in [40] for the matching of query map images.

2. Image stitching. Similar features detected in different images can be used to match and compose broader photos from the same location (they serve as reference points for the aforementioned hommography technique). In the popular work of Brown and Lowe [111], SIFT features [110] are used to automatically create panoramic representations out of an image sequence from the same site.

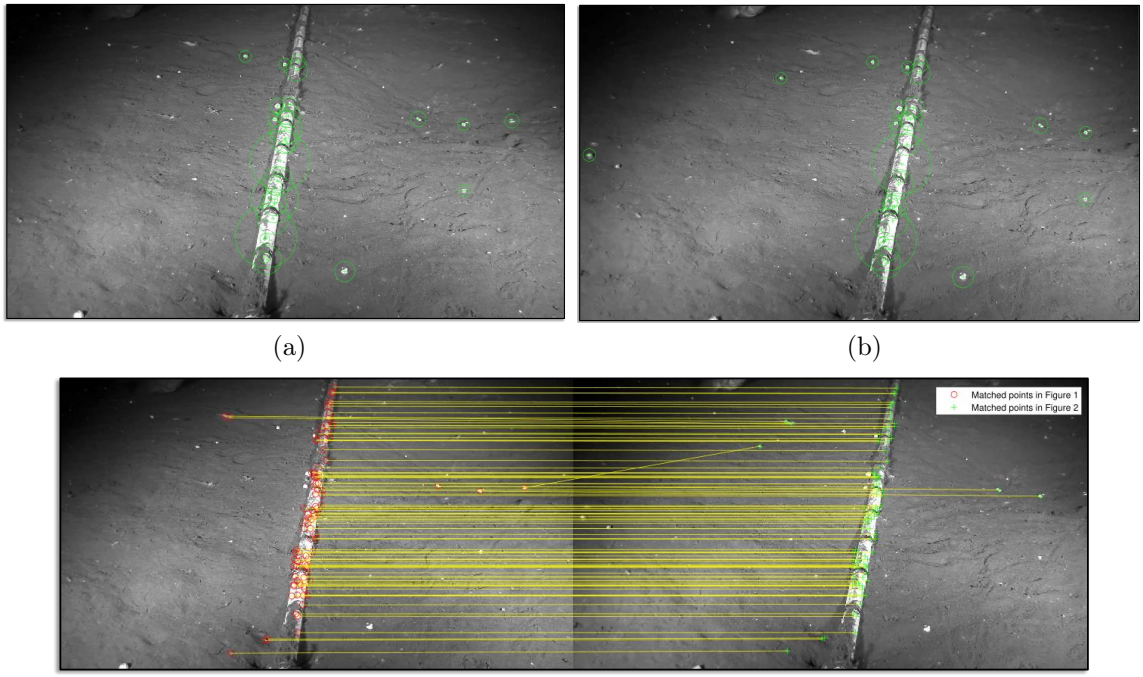
3. Photogrammetry. Extracting meaningful features from images allows photogrammetry to create maps and Digital Surface Models (DSM) of locations, as well as to provide physical measurements of them. Lingua *et al.* [53] analyzes the performance of SIFT features [110] under diverse geographical scenarios, con-

cluding that they represent suitable features for the extraction and matching of relevant information in the photogrammetry field.

Numerous other monitoring applications (*e.g.*, animal motion tracking, deforestation measurements, coastal analysis, oceanographic analysis) also make use of visual features. Optimal visual features should efficiently represent the same region under different viewpoints and transformation settings: scale, rotation, translation and illumination. In the last decade, notable feature extraction frameworks that concentrate in diverse priorities (*e.g.*, quality, performance, repeatability) were proposed. Some notable examples include SIFT [110], SURF [112], BRIEF [113], BRISK [114], ORB [115], and LIFT [116].

Figure 1.5 shows the SURF feature points (a speeded up version of SIFT that uses easy-to-compute Box Filters for scale space calculation, wavelet responses for orientation assignment, among other differences) detected in two OceanDark [70] samples from the same underwater site captured at distinct times (the camera rotated between frames). The matching points shown in Figure 1.5c can be used to determine that these images are from the same site, to perform image stitching, or even to recognize an object of interest.

Deep Learning-based image understanding. In recent years (specifically after the CNN-based AlexNet network [117] was proposed for image classification), deep convolutional neural networks have gained a lot of attention and been used to both extract meaningful features from images and perform various tasks including, but not limited to, image classification [117], object detection [118], semantic segmentation [119], instance segmentation [120], motion detection/pose estimation and tracking [121], scene reconstruction [122], style transfer [123], image colorization [124]. The popularity of these approaches lies in the fact that the visual components (features) of objects from different classes are difficult to individually study and hand-craft (so that these can be identified), especially in applications were a huge number of classes are present. For instance, in the Snapshot Serengeti dataset [89] more than 40 species of mammals have to be distinguished between each other. Conventional feature extractors/descriptors (*e.g.*, SIFT [110], SURF [112]) could be used to recognize each class, but the strong variability in the data caused by illumination and scale changes, occlusion, and inter-class differences results in an extremely challenging task. Studying and determining discriminative visual features of individual classes (*e.g.*, the antlers of an antelope or the trunk of an elephant) is an efficient approach, but it usually leads to class-specific systems that are not scalable for different viewpoints or targets.



(a) (b)

(c)

Figure 1.5: SURF features [112] detected in two OceanDark [70] samples representing distinct frames of a video. Different radius around the feature points indicate the σ corresponding to the scale in the scale space where the interest point was detected (refer to [110, 112] for details). (a) First viewpoint of an underwater site and its SURF features. (b) Second viewpoint of an underwater site and its SURF features. (c) SURF features matched between the two underwater images.

A recurrent setting posed in Environmental Monitoring becomes the following: an enormous amount of data is available (*e.g.*, Landsat imagery [18], Fish4Knowledge [81] and Snapshot Serengeti [89] datasets), and finding discriminative visual features within this imagery is paramount for its automated analysis using Computer Vision. Conventional and class-specific features have limited applicability and scalability. This is where Deep Learning can contribute the most: DL-based frameworks are capable of autonomously extracting meaningful features from the data. The data analysis process can therefore shift its focus to the collection and curation of data, development of data-driven architectures and modeling of the overall phenomena studied (instead of engineering custom features). It is important to point that features and behaviours explicitly modeled without using data-driven systems are still relevant. For instance, important visual contextual information can be directly described and considered by a system if manually included in it (*e.g.*, the DSMV module proposed

in Chapter 2).

The ability of Deep Learning-based methods to learn representations of data is also harvested in fields other than Computer Vision. Statistical distributions of a training set are learned and used to generate synthetic samples with Generative Adversarial Networks (GAN) [125]; temporal relationships (commonly between tokens such as words or letters) between data is determined by Recurrent Neural Networks (RNN) and allows for numerous Natural Language Processing (NLP) tasks [126, 127]; computational agents are trained using the DL-based representations of the data (state spaces and policies) in the Reinforcement Learning (RL) field, which would otherwise represent a difficult task for unstructured data.

Given the nature of visual Environmental Monitoring data (abundant, heterogeneous, and diverse), DL approaches have been predominantly used in recent works to aid in the study and understanding of monitored sites. Examples are the aforementioned system proposed by Norouzzadeh *et al.* [23] to recognize wild animals; the method to classify land cover and crop types from satellite imagery proposed by Kussul *et al.* [27]; Kellenberger's [47] method to detect mammals in UAV imagery, the fish species classification tool proposed by Salman *et al.* [73], as well as some of the systems proposed in this thesis [56, 76, 77, 78].

Deep learning offers powerful tools that assist in the comprehension of large amounts of data, which represents well the scenario found in Environmental Monitoring. This is the main motivation for its use in three out of the four works proposed and detailed in this thesis.

1.4 Thesis Objectives and Contributions

Visual data from Environmental Monitoring systems might take a number of different forms *e.g.*, images and videos in the visible electromagnetic spectrum, infrared images, acoustic data-based images (echograms). Given their particular composition, morphology, and scientific goals, each of these data *streams* warrants the use of a specific set of CV techniques. This thesis focuses on three different streams of data (see Table 1.2) that illustrate unique challenges that an EM setup might present.

Objectives. The objective of this thesis is to perform a thorough investigation on how to apply Computer Vision for the automatic, efficient and accurate interpretation and analysis of three different streams of Environmental Monitoring visual data. In particular, each of the projects developed in this thesis supports our hypothesis that

Table 1.2: Types of visual Environmental Monitoring-based data streams explored.

EM Data Stream	Geographic source	Characteristics & Goals
Visual (visible spectrum)	Out-of-water	Natural-looking images where marine vessels must be identified.
Visual (visible spectrum)	Underwater	Low-lighting, natural-looking images that must be enhanced for further scientific analysis.
Active acoustic	Underwater	Acoustic backscatter data visualized as images that reflect echo strength (echograms). Species (*e.g.*, salmon, herring, krill, hake) must be recognized in each sample.

even though rich and diverse, the interpretation of EM visual data benefits directly and significantly from using custom-designed, novel Computer Vision systems.

Figure 1.6 details the three Environmental Monitoring data streams considered in this thesis, as well as their particular challenges and proposed solutions. Note that there exists a number of additional scenarios for visual data in EM that would call for the use of different approaches, such as the processing of infrared imagery [31, 62, 63, 128]. However, in order to simultaneously fit the scope of this thesis and be representative, we chose a combination of data streams with fundamentally different natures, therefore requiring the development of novel and custom CV frameworks.

1. **Underwater Visual Stream** (Figure 1.6 left). This stream represents a common type of Environmental Monitoring data obtained in marine exploration studies: underwater imagery. The usage of artificial lighting sources to illuminate underwater images captured at profound depths often creates dark regions that might hide valuable biological information. The efficient analysis of such deep-water images has vital importance for different biological, geological and physical studies. The main goal associated with this data stream is therefore to enhance the levels of lighting and overall image quality (see subsection 2.4 for a group of metrics employed to evaluate this process), ultimately increasing the scientific value of these data. The images representing the underwater visual stream are a subset of years-worth of data collected by Ocean Networks Canada (ONC) using underwater cabled observatories. Chapter 2 details the computational pipeline created to enhance this stream of data.

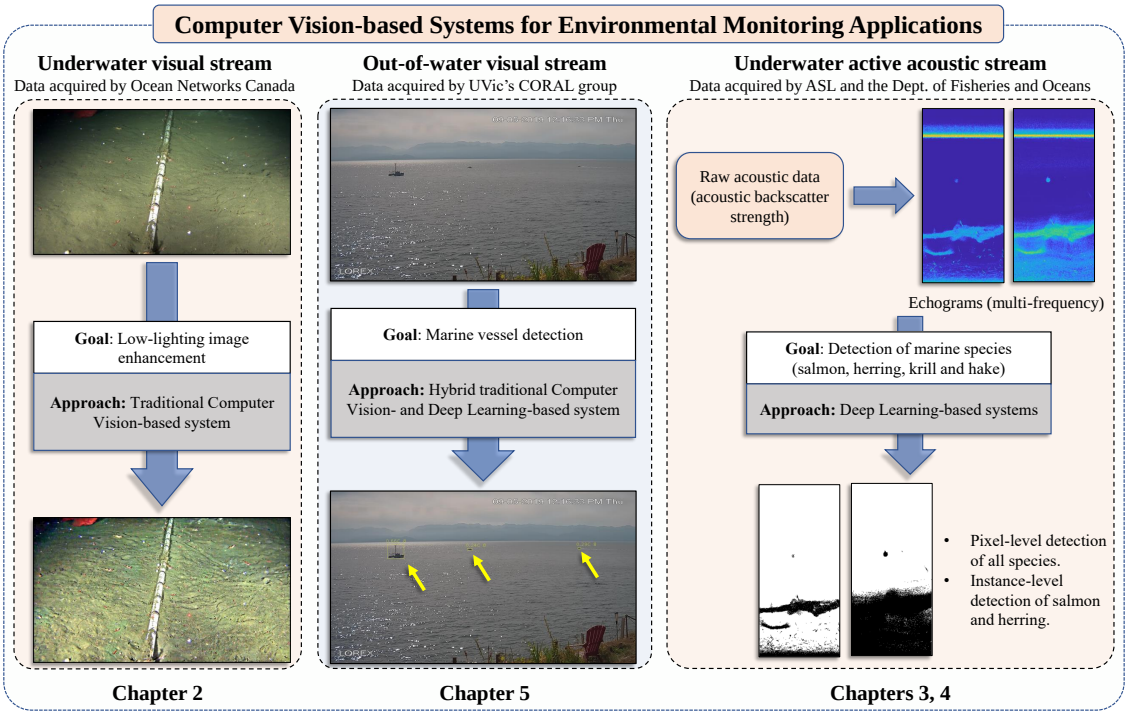


Figure 1.6: Thesis structure (the reader in invited to *zoom-in* the digital version of this Figure). **Left**: in this underwater visual stream the goal is to enhance the visibility levels (*i.e.*, darkness, contrast, sharpness) of the inputs. **Center**: considering an out-of-water visual stream, the goal in this set of visual data is to detect marine vessels of different sizes, shapes, colors and levels of visibility. **Right**: raw acoustic data acquired with echosounders are first visualized as images (echograms), followed by a pixel-level detection of diverse marine species.

2. **Out-of-water Visual Stream** (Figure 1.6 center). *In-situ* imagery is commonly captured in coastal monitoring layouts, allowing for a detailed analysis of weather conditions, marine life and traffic in the region. This visual data stream represents such scenario by using years-worth of monitoring data obtained by the Coastal and Ocean Resource Analysis Laboratory (CORAL) of the University of Victoria. Given that these data were captured in a sanctuary of Southern Resident Killer Whales (SRKW), their main interpretation goal is to efficiently identify the presence of marine vessels (which might manifest themselves as objects of different sizes, shapes, colors and levels of visibility). Chapter 5 discusses the hybrid approach (combining elements of DL and traditional CV) proposed to that end.

3. **Underwater Active Acoustic Stream** (Figure 1.6 right). This stream aims

to illustrate the diverse nature of data measured in Environmental Monitoring layouts and the ability of, given certain conditions, leveraging the capabilities of CV techniques in non-visual EM data. Active acoustic data are often captured with echosounders (*e.g.*, AZFP [129]) in underwater environments for exploration and monitoring. In this dissertation, the acoustic measurements representing this stream were collected by the Department of Fisheries and Oceans (DFO) at multiple sites across Vancouver Island, BC, Canada. By representing these raw data as 2-D matrices (echograms), one creates synthetic images that convey important information about the abundance, frequency and behaviour of multiple species of marine animals. The objective associated with this data stream is to identify four marine species: schools of salmon, herring, hake and swarms of krill. Given that these species present significantly heterogeneous morphologies in echograms (and thus require the use of specific techniques), two distinct systems were created to allow for their identification, as described in Chapters 3 and 4.

Contributions. The chief contribution of this dissertation is to perform a thorough exploration on how to innovatively apply Computer Vision within multiple Environmental Monitoring contexts. In doing so, it studies the specific requirements associated with three visual data streams and proposes efficient systems that debut, in each pair of stream/goals, numerous novel systems (as detailed in the next four Chapters).

The majority of the content included in the remaining Chapters of this thesis was peer-reviewed and presented to the scientific community in the form of journal publications and conference papers. The chronological list of publications submitted and accepted within the context of this dissertation follows:

1. **Enhancement of low-lighting underwater images using Dark Channel Prior and Fast Guided Filters** [101] *(International Conference on Pattern Recognition (ICPR): 3rd Workshop on Computer Vision for Analysis of Underwater Imagery, 2018).*

   In this work I combine the observation of Dong *et al.* [130] that darkness presents itself as haze in inverted images, and the Dark Channel Prior [131] and Fast Guided Filters [132] for the enhancement of low-lighting underwater images. Other notable contributions are the exploration of the impact of diverse choices of hyper-parameters in the transmission map filtering stage, and

the synthesis of a dataset with artificial haze.

2. **A Contrast-Guided Approach for the Enhancement of Low-Lighting Underwater Images** [70] *(MDPI Journal of Imaging, 2019).*
   In this work I tackle three fundamental problems associated with the use of single-sized patches when calculating the Dark Channel Prior (DCP) [131], transmission maps and atmospheric lighting. In order to do so I propose the Contrast Code Image (CCI), which is used for the automatic computation of the size of patches involved in the dehazing process (*i.e.*, dark channel and transmission map calculation) based on local contrast information.

3. **A Deep Learning-based Framework for the Detection of Schools of Herring in Echograms** [78] *(Conference on Neural Information Processing Systems (NeurIPS) Workshop: Tackling Climate Change with Machine Learning, 2019).*
   As an equal-contribution first author of this work, I participated in the development of a multi-stage Region of Interest (ROI) extractor, and custom-trained and explored the use of three deep learning-based image classifiers (as well as Support Vector Machines) towards the classification of schools of herring in echograms.

4. **L^{2}UWE: A Framework for the Efficient Enhancement of Low-Light Underwater Images Using Local Contrast and Multi-Scale Fusion** [69] *(Conference on Computer Vision and Pattern Recognition (CVPR) Workshop: New Trends in Image Restoration and Enhancement (NTIRE), 2020).*
   This work builds upon the CCI concept that I introduced in [70]. Considering that artificial lighting sources (common in underwater monitoring scenarios) often create non-homogeneous illumination profiles, this work uses the CCI in the calculation of DCP and two atmospheric lighting maps. Each map drives the input of a multi-scale fusion process (introduced by Ancuti *et al.* [133]), ultimately recovering an output that focuses on both preserving image details and darkness removal.

5. **Detecting Marine Species in Echograms via Traditional, Hybrid, and Deep Learning Frameworks** [77] *(International Conference on Pattern Recognition (ICPR), 2020).*
   In this work I perform a comprehensive comparison between the use of three

approaches for the identification of schools of herring: the two frameworks we debuted in [78] and a novel tilling approach that breaks inputs down into sub-regions and performs object detection (custom-trained YOLOv2 [134] and Faster R-CNN [118]) on them.

6. **Size-invariant Detection of Marine Vessels From Visual Time Series** [56] *(Winter Conference on Applications of Computer Vision (WACV), 2021).* In this work I propose a framework for the identification of marine vessels of various sizes, shapes, colors and levels of visibility in *in-situ* Environmental Monitoring visual data. The system is composed by two main modules: 1) a first stage based on pre-trained end-to-end object detectors (responsible for the identification of medium- and large-sized vessels); 2) the Detector of Small Marine Vessels (DSMV), a module that introduces a novel Gaussian Mixture Model (GMM)-based component, environmental assumptions and custom-trained image classifiers.

7. **Instance Segmentation-Based Identification of Pelagic Species in Acoustic Backscatter Data** [76] *(Conference on Computer Vision and Pattern Recognition (CVPR) Workshop: Perception Beyond the Visible Spectrum, 2021).* In this work I propose a custom-trained instance segmentation system is proposed for the identification of schools of herring and salmon. Differently from the bounding box outputs of previous works [77, 78], the proposed system described in this article provides a pixel-level identification among three classes (*i.e.*, herring, salmon and background). Results show significant improvements over the work presented in [77].

8. **Instance Segmentation of Herring and Salmon Schools in Acoustic Echograms using a Hybrid U-Net** [135] (*Conference on Robots and Vision (CRV), 2022*). As a co-author of this work, I contributed in an exploration where both data driven- and heuristics-based approaches were employed for the identification of biological targets in echograms. Environmental features such as inferred depth and solar elevation angle are used to drive a *heuristic module* that determines instances out of pixel-level detection outputs created with a U-Net-like semantic segmentation network. Experiments presented in this study show the significant performance improvements associated with the novel heuristic module proposed.

The content of Chapter 4 was submitted for publication to the IEEE Journal Of Oceanic Engineering and is currently under review:

> **U-MSAA-Net: A Multi-Scale Additive Attention-based Network for Pixel-level Identification of Finfish and Krill in Echograms**. *Submitted to the IEEE Journal of Oceanic Engineering (under review).* This work details a novel semantic segmentation network (U-MSAA-Net) I propose for the pixel-level identification of schools of finfish (mostly hake) and swarms of krill in echograms. This network debuts the Multi-Scale Additive Attention (MSAA) module, which observes feature maps of multiple scales during the decoding phase of a semantic segmentation task, effectively leveraging their complementary local and contextual information. Results on the novel FinFish and Krill (FFK) dataset attest to the efficiency of U-MSAA-Net by comparing it with several state-of-the-art DL-based frameworks.

Adjacent to the use of Computer Vision in Environmental Monitoring, during this Ph.D. program the author also second-authored three publications that focused on the detection of anomalous behavior in videos of sablefish [74, 136] and the use of Discrete Wavelet Transforms (DWT) for the lossless compression of the inputs of DL architectures [137].

The remainder of this document is organized as follows. **Chapter 2** details L^2UWE, a novel contrast-based framework created to enhance low-light underwater images representing the *underwater visual stream* of Environmental Monitoring data. The *underwater active acoustic stream* is addressed in **Chapters 3, 4**, each focusing on the proposed systems for the identification of schools of herring and salmon, and for schools of hake and swarms of krill, respectively. **Chapter 5** describes the approach proposed for detecting marine vessels in years-worth of *in-situ* monitoring images from the *out-of-water* visual data stream. **Chapter 6** finalizes this thesis by drawing conclusions and pointing to future research directions.

Chapter 2

L^2UWE: A Framework for the Efficient Enhancement of Low-Light Underwater Images Using Local Contrast and Multi-Scale Fusion

This chapter details L^2UWE, a system created for the enhancement of imagery from the *underwater visual stream*. The goal is to increase the scientific value of such EM data from both qualitative and quantitative viewpoints. L^2UWE shows that Environmental Monitoring can make use of "traditional" Computer Vision (*i.e.*, methods that are not necessarily data-driven) by leveraging physical models and novel insights about the formation of low-light underwater images.

The content of this Chapter was originally submitted, peer-reviewed in a double-blind manner, approved, and presented at the 2020 Conference on Computer Vision and Pattern Recognition (CVPR) Workshop NTIRE (New Trends in Image Restoration and Enhancement) [69]. Only small editorial modifications were done to the content of [69]. For example, given the double-blind nature of its review, the original manuscript referred to our own previous work in [138] as "Marques *et al.*"; in this chapter we specify that [138] was another contribution of this thesis. Other noteworthy modification was the addition of Algorithm 1 for clarity (originally presented in [138]).

2.1 Introduction

The study of underwater sites is important for the Environmental Monitoring field, as it provides valuable insight regarding their rich ecosystems and geological structures. Sensors placed underwater measure a host of properties, such as physical (*e.g.*, temperature, pressure, conductivity) and biological (*e.g.*, levels of chlorophyll and oxygen); visual data are captured by cameras, and acoustic data are collected with hydrophones. Cabled ocean observatories have installed and maintained various sensor layouts that allow for the continuous monitoring of underwater regions over extended time series.

Cabled ocean observatories have captured thousands of hours of underwater videos from fixed and mobile platforms [138]. The manual interpretation of these data requires a prohibitive amount of time, highlighting the necessity for semi- and fully-automated methods for the annotation of marine imagery. Mallet *et al.* [139] show that the use of underwater video cameras and their associated software in marine ecology has considerably grown in the last six decades. The efficient interpretation of underwater images requires that they maintain a certain level of *quality* (*i.e.*, contrast, sharpness and color fidelity), which is frequently not possible.

Underwater seafloor cameras often can not count on sunlight, prompting the use of artificial means of illumination. These artificial sources are not able to uniformly illuminate the entirety of a scene, and are typically employed in groups. These different and non-uniform lighting setups call for the use of specialized models of atmospheric lighting in image enhancement efforts. Low levels of lighting reduce the quality of visual data because contrast, color and sharpness are deteriorated, making it difficult to detect important features such as edges and textures. Additional challenges posed to the quality of underwater images are related to physical properties of the water medium: *absorption*, which selectively reduces the energy of the propagated light based on its wavelength, and *scattering*. The combined effect of these degradation factors results in images with dark regions, low contrast and hazy appearance.

The proposed single-image system, L^2UWE, offers a novel methodology for the enhancement of low-light underwater images. Its novelty is based on the observation that low-light scenes present particular local illumination profiles that can be efficiently inferred by considering local levels of contrast. We propose two contrast-guided atmospheric illumination models that can harvest the advantages of underwater darkness removal systems such as [138], while preserving colors and enhancing

important visual features (*e.g.*, edges, textures). By combining these outputs via a multi-scale fusion process [133] we highlight regions of high contrast, saliency and color saturation on the final result.

The performance of L^{2}UWE is compared to that of five underwater-specific image enhancers [140, 141, 142, 143, 138], as well as two low-light-specific enhancers [144, 145] using the OceanDark dataset [138]. Seven distinct metrics (*i.e.*, UIQM [146], PCQI [147], FADE [148], GCF [149], r and e-score [150], SURF features [151]) show that L^{2}UWE outperforms the compared methods, achieving a significant enhancement in overall image visibility (by reducing low-light regions) and emphasizing the image features (*e.g.*, edges, corners, textures).

2.2 Related Work

Sub-section 2.2.1 discusses works in three areas of relevance to the development of L^{2}UWE: underwater image enhancement, aerial image dehazing and low-light image enhancement. The sub-section that follows, 2.2.2, summarizes the single image dehazing framework of He *et al.* [131], the multi-scale fusion approach of Ancuti and Ancuti [133], and the contrast-guided low-light underwater enhancement system we proposed in another publication completed in the context of this thesis [138].

2.2.1 Background

Underwater image enhancement. Some early approaches that attempted the enhancement of underwater images include: the color correction scheme of Chambah *et al.* [152] that aimed for a better detection of fish, the work of Iqbal *et al.* [153], which focused in adjusting the contrast, saturation and intensity to boost images' quality, and the method of Hitam *et al.* [154], where the equalization of the histogram from underwater images is proposed as a means to achieve enhancement. More recently, Ancuti *et al.* [72] introduced a popular framework that derived two inputs (a color-corrected and a contrast-enhanced version of the degraded image), as well as four weight maps that guaranteed that a fusion of such inputs would have good sharpness, contrast and color distribution.

Multiple underwater image processing methods [155, 156, 157, 71, 158, 142, 140] make use of aerial dehazing techniques, given that the issues that plague hazy images (absorption and scattering) create outputs that are similar to those captured

underwater. Fu *et al.* [143] proposed a framework based on the Retinex model that enhances underwater images by calculating their detail and brightness, as well as performing color correction. Berman *et al.* [140] used the color attenuation and different models of water types to create a single-image underwater imagery enhancer. Cho and Kim [141], inspired by simultaneous location and mapping (SLAM) challenges, derived an underwater-specific degradation model. Drews *et al.* [142] considered only two color channels when using the dehazing approach of He *et al.* [131], adapting this method to underwater scenes. We introduced in [138] a contrast-based approach inspired in the Dark Channel Prior (DCP) [131] that significantly improved the quality of low-light underwater images.

Aerial images dehazing. Dehazing methods aim for the recovery of the original radiance intensity of a scene and the correction of color shifts caused by scattering and absorption of light by fluctuating particles and water droplets. Initial approaches proposed to address this challenge [159, 160, 161, 162, 163] were limited by the need of multiple images (captured under different weather conditions) as input. He *et al.* [131] proposed a popular method for single-image dehazing that introduced the Dark Channel and the Dark Channel Prior, which allows for the estimation of the transmission map and atmospheric lighting of a scene, both important parameters of the dehazing process (as detailed in sub-section 2.2.2). Comparative results [164, 165, 166, 138] attest to the performance of this method in scenarios where only a single hazy image is available. Numerous underwater-specific enhancement frameworks [138, 142, 167, 155, 168, 169] are based on variations of the DCP. Recently a number of data-driven methods [170, 165, 171, 172, 173, 174] focused on the use of Convolutional Neural Networks (CNN) to train systems capable of efficiently performing the dehazing/image recovering task. However, these systems typically require time-consuming processes of data gathering and curation, hyper-parameter tuning, training, and evaluation.

Low-light image enhancement. Dong *et al.* [130] observed that dark regions in low-light images are visually similar to haze in their inverted versions. The authors proposed to use the DCP-based dehazing method to remove such haze. Ancuti *et al.* [175] proposed to use a non-uniform lighting distribution model and the DCP-based dehazing method to generate two inputs (an additional input is the Laplacian of the original image), followed by a multi-scale fusion process. Zhang *et al.* [145] introduced the maximum reflectance prior, which states that the maximum local intensity in low-light images depends solely on the scene illumination source. This

prior is used to estimate the atmospheric lighting model and transmission map of a dehazing process (refer to sub-section 2.2.2 for details). Guo *et al.* [144] proposed LIME, where the atmospheric lighting model is also initially constructed by finding the maximum intensity throughout the color channels. The LIME framework refines this initial model by introducing a structure prior that guarantees structural fidelity to the output, as well as smooth textural details. Data-driven approaches were also proposed for the enhancement of low-light images [176, 177, 178, 179]. These methods employ CNNs to extract features from datasets composed of low-light/normal-light pairs of images ([179] requires only the degraded images), and then train single-image low-light enhancement frameworks.

2.2.2 Previous Works Supporting the Proposed Approach

Dark Channel Prior-based dehazing of single images. Equation 2.1 describes the formation of a hazy image I as the sum of two additive components, *direct attenuation* and *airlight*.

$$I(x) = J(x)t(x) + A_{\infty}(1 - t(x)) \tag{2.1}$$

where J represents the haze-less version of the image, x is an spatial location, transmission map t indicates the amount of light that reaches the optical system and A_{∞} is an estimation of the global atmospheric lighting. The first term, direct attenuation $D(x) = J(x)t(x)$, indicates the attenuation suffered by the scene radiance due to the properties of the medium. The second term, airlight $V(x) = A_{\infty}(1 - t(x))$, is due to previously scattered light and may result in color shifts on the hazy image. The goal of dehazing efforts is to find the haze-free version of the image (J) by determining A_{∞} and t.

He *et al.* [131] introduced the Dark Channel and the Dark Channel Prior, which can be used to infer atmospheric lighting A_{∞} and to derive a simplified formula for the calculation of t. The dark channel is described in Equation 2.2.

$$J^{dark}(x) = \min_{y \in \Omega(x)} (\min_{c \in \{r,g,b\}} I^{c}(y)) \tag{2.2}$$

where x and y represent two pairs of spatial coordinates in the dark channel J^{dark} and in the hazy image I (or any other arbitrary image), respectively. The intensity of each pixel in the single-channel image J^{dark} is the lowest value between the pixels

inside patch Ω in I^c, where $c \in \{R, G, B\}$. The DCP is an empirical observation stating that pixels from non-sky regions have at least one significantly low intensity across the three color channels. Thus the dark channel is expected to be mostly dark (*i.e.*, intensities closer to 0).

A single three-dimensional global atmospheric lighting vector A^c_∞ ($c \in \{R, G, B\}$) can be inferred by looking at the 0.1% [131] or 0.2% [138] brightest pixels in the dark channel (which represent the most haze-opaque regions from input), then considering the brightest intensity pixels in these same spatial coordinates from the *input* image I. Ancuti *et al.* [175] observed that a single global value might not properly represent the illumination of low-light scenes, thus proposing the estimation of local atmospheric light intensities $A^c_{L\infty}$ inside patches Ψ following Equation 2.3.

$$A^c_{L\infty}(x) = \max_{y \in \Psi(x)} (\min_{z \in \Omega(y)} (I^c(z))) \tag{2.3}$$

where x, y and z represent, respectively, spatial coordinates in the local atmospheric image, "minimum" image $I_{\min}(z) = \min_{z \in \Omega(y)}(I(z))$, and hazy image I. For each color channel $c \in \{R, G, B\}$, the local atmospheric lighting $A^c_{L\infty}$ is calculated by first finding $I^c_{\min}$, which represents the minimum intensities inside patch Ω on I^c, and then calculating the maximum intensities inside a patch Ψ on $I^c_{\min}$. By arguing that the influence of lighting sources goes beyond patch Ω, Ancuti *et al.* [175] used patch Ψ with twice the size of Ω. The DCP is used in [131] to derive Equation 2.4 for the calculation of the transmission map t.

$$t(x) = 1 - \omega \min_{y \in \Omega(x)} (\min_{c \in \{r,g,b\}} \frac{I^c(y)}{A^c_\infty}) \tag{2.4}$$

where A^c_∞ indicates the atmospheric lighting in the range $[0, 255]$, resulting in a normalized image ($\frac{I^c(y)}{A^c_\infty}$) ranging from $[0, 1]$. The constant ω ($0 \leq \omega \leq 1$) preserves a portion of the haze, generating more realistic outputs. Note that for the local estimation of atmospheric lighting, A^c_∞ would have to be substituted by $A^c_{L\infty}$ in Equation 2.4. The haze-less version of the image, $J(x)$, is obtained using Equation 2.5 [131].

$$J^c(x) = \frac{I^c(x) - A^c_\infty}{\max(t(x), t_0)} + A^c_\infty \tag{2.5}$$

Since the direct attenuation $D(x) = J(x)t(x)$ can be close to zero when $t(x) \approx 0$, [131] adds the t_0 term as a lower bound to $t(x)$, effectively preserving small amounts

of haze in haze-dense regions of I.

Contrast-guided approach for the enhancement of low-light images. In [138] we observed and addressed three problems that arrive from the use of single-sized patches Ω throughout the image dehazing process: 1) small patch sizes result in oversaturation of the radiance from the recovered scene (non-natural colors); 2) large patch sizes better estimate and eliminate haze, but since they consider that the transmission profile (*i.e.*, amount of light that reaches a camera) is constant inside patch Ω, undesired halos are might emerge around intensity discontinuities and 3) a single patch size is typically not optimal for images of varying scales.

We argued in [138] that homogeneous regions of an image possess lower contrast, and that the assumption that their transmission profile is approximately the same holds for patches Ω of varying sizes (in particular, from 3×3 up to 15×15). In these scenarios, the use of larger patch sizes generates darker dark channels (strengthening the assumption of the DCP) and are, therefore, preferred. For regions with complex content (*i.e.*, intensity changes), we propose the use of smaller patch sizes to capture the local transmission profile nuances. To determine the correct patch size for each pixel in image I, we introduced in [138] the *Contrast Code Image (CCI)*, whose calculation is detailed in Algorithm 1 (note that two-parameter spatial coordinates (x, y) are used in this Algorithm for clarity, in contrast with the single-parameter representation of coordinates in the rest of the Chapter).

To determine the CCI of an image, we study the standard deviation σ inside different-sized patches centered at the same pixel coordinate in the hazy image. If varying the patch size results in an increase of σ, the variation is undesired. Given that the most commonly used patch sizes observed in the literature range from 3×3 to 15×15 pixels [180] (and also that patch sizes larger than that significantly increase processing time), seven different options of patch sizes are considered in the contrast-guided approach: 3×3, 5×5, 7×7, 9×9, 11×11, 13×13 and 15×15.

The seven values of σ for each spatial coordinate (x, y) in the input image are calculated and the lowest one is stored in the same position (x, y) of the two-dimensional matrix *Sigmas*. A second two-dimensional matrix, *CCI*, stores an integer code c $(1 \leq c \leq 7)$ referring to the patch size that resulted in the smallest σ for each coordinate (x, y). Different codes c refer, respectively, to patches sized from 3×3 up to 15×15.

Since bigger patch sizes are preferred in order to strengthen the DCP (larger patches increase the likelihood of encompassing a near-zero-valued pixel), all codes

c in the CCI are initially populated with value 7. To further stimulate the usage of bigger patch sizes, an additional parameter, tolerance t, is introduced. This parameter represents the percentage by which a subsequent σ has to be smaller than the previous one in order to trigger a new code c to be stored in the CCI.

Algorithm 1: Calculation of the contrast code image (CCI).

Data: Hazy image
Result: Contrast code image
$CCI \leftarrow$ zeros
$Sigmas \leftarrow$ zeros
while *not all pixels (x,y) in hazy image are visited* **do**
  $c \leftarrow 7$
  while $c >= 1$ **do**
    $s \leftarrow (c * 2) + 1$
    $std \leftarrow$ standard deviation inside $s \times s$ square window centered at pixel (x, y)
    if $c = 7$ **then**
      $CCI(x, y) \leftarrow 7$
      $Sigmas(x, y) \leftarrow std$
    else
      $decay \leftarrow 1 - (t/100)$
      if $std < (Sigmas(x, y) * decay^{c-1})$ **then**
        $CCI(x, y) \leftarrow c$
      end
    end
    $c \leftarrow c - 1$
  end
end

The calculation of the CCI is summarized in Equation 2.6.

$$CCI(x) = \arg\min_{i}[\sigma(\Omega_i(x))] \tag{2.6}$$

where $\Omega_i(x) \in I$ represents a square patch of size $(2i + 1) \times (2i + 1)$ centered at spatial coordinates x, $i = \{1, 2, ..., 7\}$.

We propose in [138] to use the CCI as a guiding parameter for the calculation of the transmission map and dark channel. For a pixel location x, one would use patches

of size $(2c\ +\ 1) \times (2c\ +\ 1)$ (where $c = CCI(x)$) instead of fixed-size patches. This contrast-guided approach significantly mitigates the three aforementioned problems.

Multi-scale fusion for image enhancement. The work of Ancuti *et al.* [133] proposes to perform dehazing by first calculating two inputs $\mathcal{I}^k$ ($k = \{1, 2\}$) from the original image: a white-balanced and a contrast-enhanced version of it. Then the authors introduced the now-popular multiscale fusion process, where three *weight maps* containing important features of each input $\mathcal{I}^k$ are calculated: 1) **Luminance** $\mathcal{W}_L^k$: responsible for assigning high values to pixels with good visibility, this weight map is computed by observing the deviation between the R, G and B color channels and luminance L (average of pixel intensities at a given location) of the input; 2) **Chromaticity** $\mathcal{W}_C^k$: controls the saturation gain on the output image, and can be calculated by measuring the standard deviation across each color channel from the input; 3) **Saliency** $\mathcal{W}_S^k$: in order to highlight regions with greater saliency, this weight map is obtained by subtracting a Gaussian-smoothed version of the input by its mean value (method initially proposed by Achanta *et al.* [181]). Aiming to minimize visual artifacts introduced by the simple combination of the weight maps, [133] uses a multiscale fusion process where a Gaussian pyramid is calculated for the normalized weight map $\bar{\mathcal{W}}^k$ (*i.e.*, per-pixel division between the multiplication of all three weights and their sum) of each input, while the inputs $\mathcal{I}^k$ themselves are decomposed into a Laplacian pyramid [182]. Given that the number of levels from both pyramids is the same, they can be independently fused using Equation 2.7 [133].

$$\mathcal{R}_l(x) = \sum_k G_l\{\bar{\mathcal{W}}^k(x)\} L_l\{\mathcal{I}_k(x)\} \tag{2.7}$$

where l indicates the number of levels (typically 5), $L\{\mathcal{I}\}$ represents the representation in the Lapacian pyramid of $\mathcal{I}$, and $G\{\bar{\mathcal{W}}\}$ denotes the Gaussian version (*i.e.*, representation in different scales of the Gaussian pyramid) of $\bar{\mathcal{W}}$. The fused result is a sum of the contributions from different layers, after the application of an appropriate upsampling operator.

The authors of [133] applied the multi-scale fusion with different strategies in other works, for example by changing the inputs to gamma-corrected and sharpened versions of a white-balanced underwater image [71] or by calculating new weight maps (*e.g.*, local contrast weight map [175]).

2.3 Proposed Approach

Although we obtained good enhancement results for low-light underwater images in [138], the approach proposed in that work oversimplifies the scene atmospheric lighting by using a single 3-channel value A_∞ for the entire image. This common practice, which assumes a mostly white-colored lighting source, works well with aerial hazy images under sunlight [131], but fails to represent low-light scenarios properly [175], since those can present non-homogeneous and non-white illumination. As a result, the enhanced images we obtained in [138] present regions that suffer an excess of darkness removal, generating outputs that are overly bright and have a washed-out appearance, often belittling intensity discontinuities (*e.g.*, edges, textures) that could represent important visual features in other computer vision-based applications. This undesirable phenomenon is also reflected in the low performance results of the method we proposed in [138] for some enhancement metrics (e-score [150] and count of SURF [151] features).

With L^{2}UWE we propose a better image enhancement mechanism by deriving more realistic models for underwater illumination. We consider local contrast information as a guiding parameter to derive lighting distribution models under two distinct "lenses": one very focused (*i.e.*, using smaller local regions) that captures the finer details of the original image, and a second, wider one, which considers larger local regions to create brighter illumination models. Each model drives a different dehazing process, and the outputs are combined using a multi-scale fusion approach. This fusion strategy preserves both the details and darkness removal obtained with the two lighting models. As a result, the output of L^{2}UWE drastically reduces the amount of darkness from the original images while highlighting their intensity changes. Figure 2.1 details the computational pipeline of L^{2}UWE.

2.3.1 Contrast-aware Local Atmospheric Lighting Models for Low-light Scenes

Using parts of the dark channel to derive a single *global estimate* A_∞ for underwater images creates overly-bright and washed-out results. Our underlying assumption in [138] that the lighting in inverted input images is mostly white is not reasonable for underwater scenes: see the "inversion" step of Figure 2.1, where the lighting presents non-white colors. Given that the images from our OceanDark dataset [138] are cap-

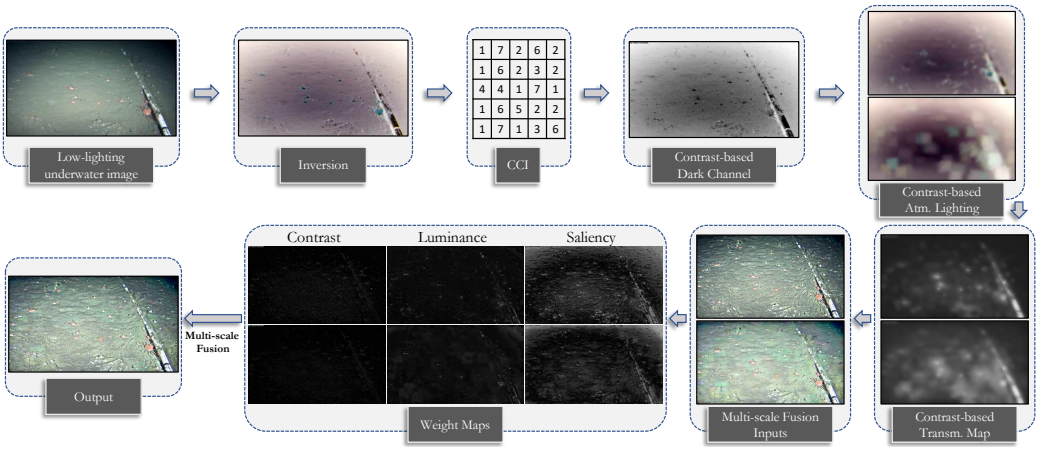


Figure 2.1: Pipeline of L^{2}UWE. First, the *CCI* [138] is calculated in the inverted version of the image. The contrast-based Dark Channel is then used to derive two atmospheric lighting models that consider local illumination. Two transmission maps generate the inputs of a multi-scale fusion process. Three weight maps are calculated for each input, which are then combined to offer the framework's output.

tured using artificial (and often multiple) lighting sources, a single global estimate A_∞ can not properly model their non-uniform nature. L^{2}UWE addresses these problems by calculating the atmospheric lighting for each color channel, and by considering the CCI code at each spatial coordinate x when determining *local estimates* of lighting. This approach is similar to that described in Equation 2.3 [175], but instead of using a fixed-size patch Ψ, we introduce the contrast-guided patch Υ for lighting calculation.

We observed that regions with high contrast (*i.e.*, lower contrast code c in the CCI) should have their local lighting component modeled by considering a larger Υ, based on the heterogeneous influence that illumination sources place on them. On the other hand, since regions with lower contrast (*i.e.*, higher contrast codes c in the CCI) are illuminated in a roughly homogeneous manner, only smaller patches Υ need to be observed to properly model local illumination. We offer a formalization of this reasoning in Equation 2.8, which specifies the relationship between CCI code c and the size of the lighting square patch Υ used in our local atmospheric lighting model calculation.

$$S_\Upsilon(m, c) = 3m - [\frac{m}{3}(c - 1)] \tag{2.8}$$

where m represents an arbitrary multiplication factor and $c = \{1, 2, ..., 7\}$ represents code c in the CCI at a certain spatial coordinate. Parameter m has a multiplicative effect on the contrast-guided patch, but an offset is also added to progressively con-

strain it based on the patch size: smaller patches will be more influenced than larger patches. For example, for $m = 15$ and a coordinate x where $CCI(x) = 1$, patch $\Upsilon(x, m)$ would be of dimensions $S_\Upsilon(15, 1) \times S_\Upsilon(15, 1)$, or 45×45 pixels. Similarly, for $m = 15$ and $CCI(x) = 5$ (lower contrast region), patch $\Upsilon(x, m)$ would be of dimensions 25×25 pixels.

With the use of contrast-aware patch Υ and multiplication factor m, we define the local, contrast-guided atmospheric intensity $A^c_{LCG\infty}$ for a color channel c as:

$$A^c_{LCG\infty}(x, m) = \max_{y \in \Upsilon(x,m)} (\min_{z \in \Omega(y)} (I^c(z))) \tag{2.9}$$

The main distinction between $A^c_{LCG\infty}(x, m)$ (Equation 2.9) and $A^c_{L\infty}(x)$ (Equation 2.3) is that the former uses contrast-aware patches $\Upsilon(x, m)$, while the latter uses fixed-size patches $\Psi(x)$ (we thus refer the reader to Equation 2.3 and its discussion on sub-section 2.2.2). Since $A^c_{LCG\infty}$ is calculated for each color channel c, by maximizing contrast-aware patch $\Upsilon(x, m)$ one is actually determining a local position y where the radiance for a certain color channel c is maximum in the I^c_{min} (*i.e.*, a dark channel specific for color channel c). Figure 2.2 compares the different atmospheric lighting models obtained using $A_{LCG\infty}$ and A_∞. The $A_{LCG\infty}$ is filtered using a Gaussian kernel with $\sigma = 10$ to prevent the creation of abrupt, square-like intensity discontinuities ("halos") when normalizing the input image by the atmospheric lighting (see Equation 2.4). Differently from A_∞, $A_{LCG\infty}$ captures the color characteristics of the illumination, as well as its local distribution throughout the image.

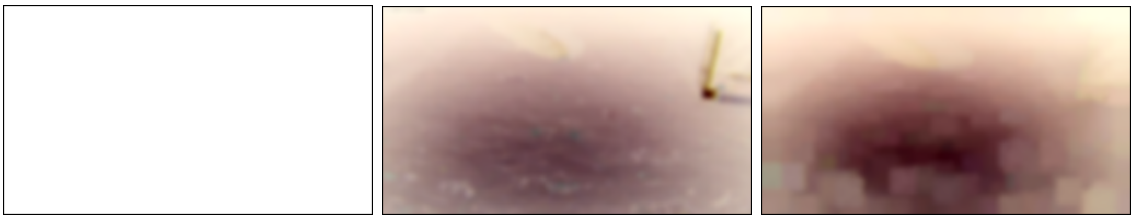


Figure 2.2: Different approaches for atmospheric lighting estimation on inverted OceanDark [138] sample. **Left:** a single 3-channel value (A_∞), implying that the illumination is homogeneous and roughly white. **Center:** contrast-aware local lighting estimation $A^c_{LCG\infty}$. **Right:** Gaussian-smoothed version of $A^c_{LCG\infty}$, mitigating the creation of halos in the enhanced output.

2.3.2 Fusion process

The choice of parameter m in Equations 2.8 and 2.9 determines the size of local window Υ and therefore the radius of influence from each illumination source. In other words, higher m creates brighter lighting models. While this generates output with less darkness, it also might apply an excess of radiance correction (*i.e.*, bringing intensities close to saturation) in regions of the image and hide important intensity changes. In order to harvest the advantages of both approaches, we derive two $A_{LCG\infty}$: one with $m = 5$, a second one with $m = 30$. These lighting models are used with Equation 2.4 to determine two transmission maps. These maps are filtered using a Fast Guided filter [132], and finally used to recover two haze-less versions of the original image (Equation 2.5). The image enhancement results obtained using each $A_{LCG\infty}$ serve as inputs $\mathcal{I}^k$ ($k = \{1, 2\}$) to a multi-scale fusion process. Figure 2.3 shows two inputs generated with $m = \{5, 30\}$ for an OceanDark [138] sample.

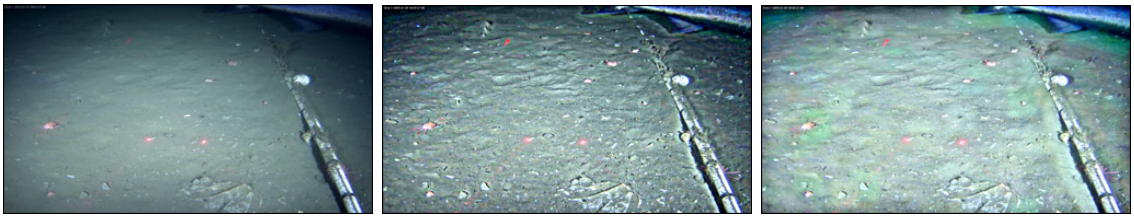


Figure 2.3: Multi-scale process inputs. **Left:** original OceanDark [138] sample. **Center:** first input, an image enhanced using the contrast-guided CCI and $A_{LCG\infty}$ with $m = 5$. **Right:** second input, an image enhanced using the contrast-guided CCI and $A_{LCG\infty}$ with $m = 30$.

By fusing the inputs generated from different $A_{LCG\infty}$, we preserve two important aspects of the enhanced images: the efficient darkness removal of the input obtained with $m = 30$ (Figure 2.3 right) and the edges, textures and overall intensity changes of the input generated with $m = 5$ (Figure 2.3 center). Experiments evaluating different pairs of m values yielded the best performance when using $m = 5$ and $m = 30$. We also performed tests using a higher number of inputs, but similar results were obtained as for fusing the inputs corresponding to $m = 5$ and $m = 30$, at the expense of a higher computational complexity.

In order to properly combine the two inputs, we chose to calculate three weight maps from each of them: saliency, luminance and local contrast. These weight maps guarantee that regions in the inputs that have high saliency and contrast, or that possess edges and texture variations will be emphasized in the fused output [133, 175].

As discussed in sub-section 2.2.2, the **saliency weight map** is calculated by subtracting a Gaussian-smoothed version of input k, $\mathcal{I}_k^{G_s}$, by the mean intensity value of this same input, $\mathcal{I}_k^{\mu}$ (constant for each input), as detailed in Equation 2.10.

$$\mathcal{W}_S^k(x) = \| \mathcal{I}_k^{G_s}(x) - \mathcal{I}_k^{\mu} \| \tag{2.10}$$

where x represents a spatial coordinate of input k and $\mathcal{I}_k^{G_s}$ is obtained with a 5×5 ($\frac{1}{16}[1, 4, 6, 4, 1]$) Gaussian kernel.

Considering that saturated colors present higher values in one or two of the R, G, B color channels [133], we use Equation 2.11 to calculate the **luminance weight map** $\mathcal{W}_L^k$.

$$\mathcal{W}_L^k = \sqrt{\frac{1}{3}[(R^k - L^k)^2 + (G^k - L^k)^2 + (B^k - L^k)^2]} \tag{2.11}$$

where L^k represents, at each spatial position, the mean of R, G, B intensities for input k. R^k, G^k and B^k are the three color channels of input $\mathcal{I}^k$.

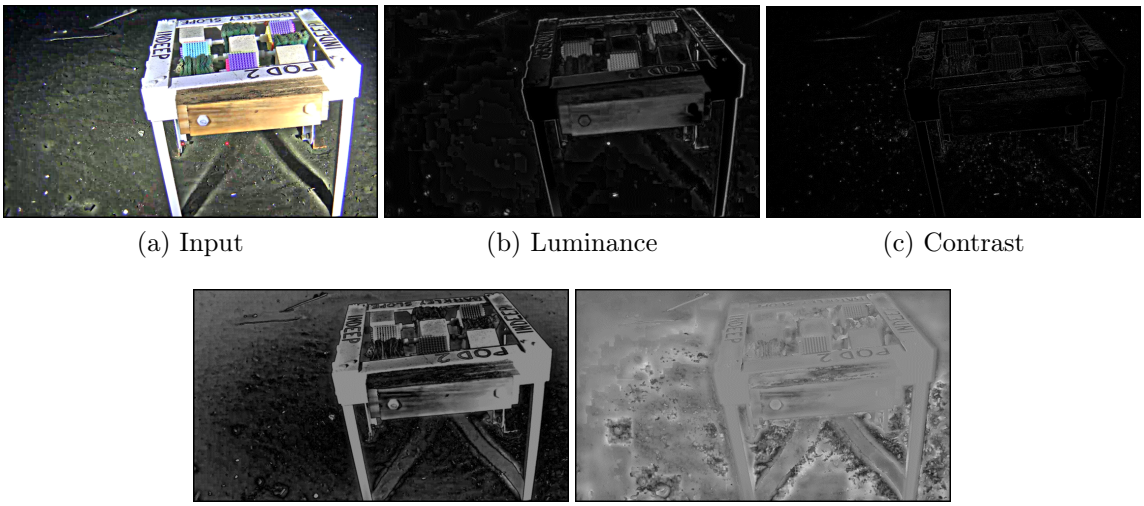
The **local contrast weight map** $\mathcal{W}_{LCon}^k$, also used in [175], is responsible for highlighting regions of input $\mathcal{I}^k$ where there exists more local intensity variation. We calculate this map by applying a $\frac{1}{8}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ Laplacian kernel on L^k.

The three weight maps ($\mathcal{W}_{LCon}^k$, $\mathcal{W}_L^k$, $\mathcal{W}_S^k$) are combined into a normalized weight map $\bar{\mathcal{W}}^k$, from which a 5-level Gaussian pyramid $G\{\bar{\mathcal{W}}^k\}$ is derived. Our choice of using Gaussian pyramids is based on their efficacy in representing weight maps, as demonstrated by Ancuti*et al.* [133]. Figure 2.4 illustrates the three weight maps calculated for one input image, as well as their corresponding normalized weight map.

Following the procedure of [175, 133, 71, 183], each input $\mathcal{I}^k$ is decomposed into a 5-level Laplacian pyramid $L\{\mathcal{I}^k\}$. The multi-scale fusion process is then carried out using Equation 2.7. The output of L^{2}UWE (Figure 2.5d) reduces the darkness from the original image, retains colors and enhances important visual features.

2.4 Experimental Results and Discussion

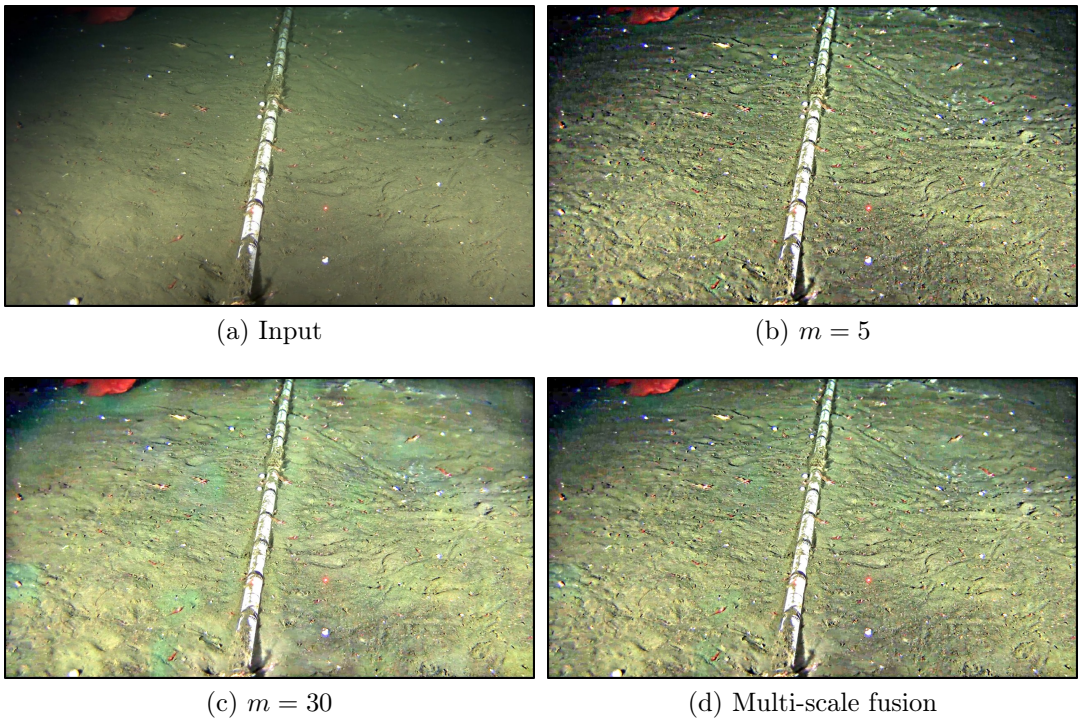
The performance of L^{2}UWE is evaluated using our OceanDark dataset [138], composed by 183 samples of low-light underwater images captured by Ocean Networks Canada (ONC). Seven metrics are used in a comprehensive comparison between L^{2}UWE and five underwater-specific image enhancers, as well as two low-light-specific



(a) Input (b) Luminance (c) Contrast

(d) Saliency (e) Normalized combination

Figure 2.4: Weight maps of an input $\mathcal{I}^k$. **(a)** Input image. **(b)** Luminance weight map $\mathcal{W}_L^k$. **(c)** Local contrast weight map $\mathcal{W}_{LCon}^k$. **(d)** Saliency weight map $\mathcal{W}_S^k$. **(e)** Normalized weight map $\bar{\mathcal{W}}^k$.



(a) Input (b) $m = 5$

(c) $m = 30$ (d) Multi-scale fusion

Figure 2.5: Enhancement results of the proposed system. **(a)** Original sample from OceanDark. **(b)** Input calculated with $m = 5$. **(c)** Input calculated with $m = 30$. **(d)** Final output of L^{2}UWE, a multi-scale fusion between the two inputs.

image enhancers.

Metrics. Evaluating the results of image enhancers in datasets without ground truth (such as OceanDark [138]) is non-trivial. Individual metrics can not, alone, indicate the performance of the enhancement; *e.g.*, the *e*-score [150] measures the number of new visible edges obtained after enhancement, which can represent noise, while the FADE score [148] can be used to measure darkness, but it does not account for visual features lost because of over-illumination. Thus we chose to employ a group of seven metrics in the evaluation. **UIQM** [146]: inspired on the human visual system, this no-reference underwater image quality indicator combines measures of colorfulness, sharpness and contrast; **PCQI** [147]: a method developed to assess the quality of contrast-changed images, PCQI considers local contrast quality maps; **GCF** [149]: this metric reflects the level of contrast present in the whole image (dehazed images typically present higher contrast); **e-** and **r-scores** [150]: indicate, based on the original and enhanced images, the increase in number of visible edges, and the boost in gradient value for these edge's pixels ("visibility"), respectively; **FADE** [148]: measures the amount of perceived fog (or darkness in inverted images), thus assigning, in our analysis, lower values to better-illuminated images; **SURF** [151]: a popular method that extracts useful features for image matching, reconstruction, stitching, object detection, among others (see sub-section 1.3.1 for details).

Comparison with state-of-the-art methods. We evaluate L^{2}UWE against five frameworks designed specifically for the enhancement of underwater images: our own previous work [138], Berman *et al.* [140], Drews *et al.* [142], Fu *et al.* [143] and Cho *et al.* [141]. Since OceanDark [138] is composed of low-light images, we also considered two popular low-light-specific image enhancers: Zhang *et al.* [145] and Guo *et al.* [144]. All methods are discussed in sub-section 2.2.1. The implementations used are those made publicly available by the authors.

Qualitative analysis. Figure 2.6 shows that L^{2}UWE is able to generate output images that highlight important visual features (*e.g.*, fish that were not visible, small rocks and overall geography of the sites) of the original images without excessively brightening the scenes. While our method from [138] and Guo *et al.* [144] yielded similar results that greatly reduced low-light regions, they also concealed important visual features because of a lighting over-correction: close-to-saturation pixel intensities might hide the finer details of the image, such as edges and textures. The methods of Drews *et al.* [142], Zhang *et al.* [145] and Cho *et al.* [141] actually darkened the images, contrary to the goal of the enhancement. The methods of Zhang *et al.* [145]

and Fu *et al.* [143] highlighted edges and textures of the outputs, but were not able to properly illuminate dark regions. The unnatural colors generated by the method of Berman *et al.* [140] can be attributed to the automatic, but often sub-optimal, choice of water type done for each sample.

Quantitative analysis. Table 2.1 details the mean values obtained by each of the methods (considering all samples of the OceanDark [138] dataset) for aforementioned set of seven metrics. L^{2}UWE obtained the highest score on the UIQM [146] metric, confirming the human-visual-system-inspired perceived image quality of its outputs. The quality of the contrast modification measured by PCQI [147] also favors the results obtained with the proposed method. The framework of Zhang *et al.* [145] obtained the highest GCF [149], however a qualitative analysis of its results indicates that this is due to the strong saturation of the output, which creates a high *global* contrast with respect to the undesired dark regions still present in the enhanced images. The highest increase in visible edges (e-score [150]) and SURF features [151] is attributed to L^{2}UWE (we reiterate that these two metrics should be interpreted cautiously, as they may reflect noise introduced to the image). Finally, the highest scores in the metrics related to visibility (r-score [150]) and FADE [148] indicate that L^{2}UWE successfully achieved the goal of increasing the illumination of low-light underwater images while preserving their colors and highlighting their salient visual structures.

Method	UIQM [146]	PCQI [147]	GCF [149]	e-score [150]	r-score [150]	FADE [148]	SURF [151]
Original	0.88 ± 0.13	1	3.28 ± 0.62	N/A	1	1.91 ± 0.79	340 ± 293
Marques [138]	0.99 ± 0.12	0.85 ± 0.03	3.41 ± 0.71	0.28 ± 0.32	2.75 ± 0.76	0.46 ± 0.18	705 ± 470
Berman [140]	1.00 ± 0.18	0.78 ± 0.15	3.84 ± 1.07	0.25 ± 0.50	2.91 ± 1.96	1.15 ± 0.40	425 ± 317
Fu [143]	0.93 ± 0.15	0.93 ± 0.09	3.28 ± 0.57	0.09 ± 0.39	1.72 ± 0.35	1.75 ± 0.25	865 ± 478
Cho [141]	1.24 ± 0.15	0.87 ± 0.04	4.11 ± 0.84	0.89 ± 0.54	1.72 ± 0.09	1.81 ± 0.52	751 ± 428
Drews [142]	1.38 ± 0.14	0.85 ± 0.05	4.70 ± 0.89	1.06 ± 0.83	1.29 ± 0.31	2.08 ± 0.90	589 ± 324
Zhang [145]	1.28 ± 0.08	1.03 ± 0.07	$\mathbf{6.34 \pm 0.74}$	1.48 ± 0.88	3.70 ± 1.00	0.72 ± 0.39	1719 ± 620
Guo [144]	0.93 ± 0.13	0.86 ± 0.03	3.50 ± 0.73	0.16 ± 0.14	2.21 ± 0.64	0.60 ± 0.24	607 ± 452
L^{2}UWE	$\mathbf{1.38 \pm 0.11}$	$\mathbf{1.17 \pm 0.07}$	4.89 ± 0.66	$\mathbf{1.82 \pm 0.83}$	$\mathbf{4.61 \pm 0.58}$	$\mathbf{0.42 \pm 0.20}$	$\mathbf{1856 \pm 655}$

Table 2.1: Mean and standard deviation of seven metrics computed on all samples of the OceanDark dataset [138]. The results compare the output of diverse image enhancing methods. Best results (*i.e.*, higher, with the exception of darkness-indicating FADE [148]) are bolded.

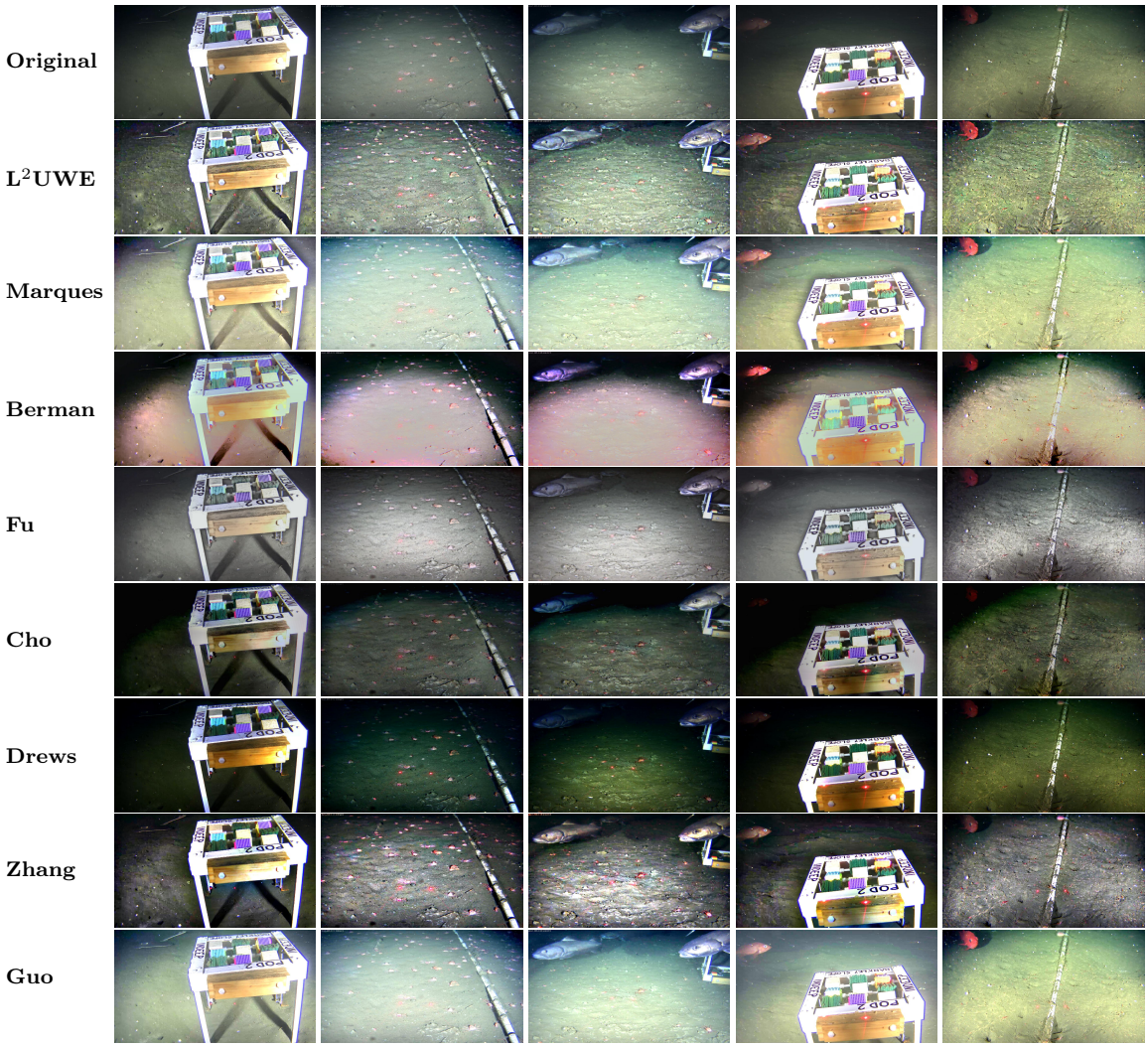


Figure 2.6: Output of diverse image enhancement methods on samples from the OceanDark dataset [138].

2.5 Conclusion

The proposed single-image enhancement framework, L^{2}UWE, uses a contrast-guided approach for the efficient modelling of lighting distributions in low-light underwater scenes. It then generates two dehazed inputs that are combined employing a multi-scale fusion process, ultimately reducing dark regions and highlighting important visual features of the original image without changing its color distribution.

Experimental results show the capacity of the proposed method of drastically reducing low-light regions of inputs without creating washed-out outputs (see Figure 2.6). Although other methods can be used for the enhancement of similar scenarios with remarkable results, our experiments show that L^{2}UWE outperforms the current state-of-the-art approaches in the task of enhancing low-light underwater images. Our proposed contrast-guided computational pipeline is also expected to work well in other mediums, such as night-time and low-light aerial scenes.

Future developments of this framework will focus on the adaptation of L^{2}UWE for aerial low-light images, its evaluation in additional datasets, and the use of data-driven systems in the framework's computational pipeline (*e.g.*, a CNN-based network for the estimation of atmospheric lighting models).

2.6 Acknowledgments

The work described in this Chapter was completed with generous in-kind support (data, infrastructure, expertise) from Ocean Networks Canada.

Chapter 3

Instance Segmentation-Based Identification of Pelagic Species in Acoustic Backscatter Data

This chapter proposes a Deep Learning-based system for the timely and effective identification, on a pixel level, of schools of herring and salmon in acoustic backscatter images (echograms). Aside from its direct biological and acoustic goals, this work also aimed to explore the usefulness of Computer Vision in Environmental Monitoring visual data, even in scenarios where non-natural-images are provided. As a result, it supports the exploration of the second EM data stream described in Table 1.2, *underwater active acoustic*. This is the third publication of this project [78, 184, 76] completed by the author under the present thesis.

The content of this Chapter was originally submitted, peer-reviewed in a double-blind manner, approved, and presented at the 2021 Conference on Computer Vision and Pattern Recognition (CVPR) Workshop PBVS (Perception Beyond the Visible Spectrum) [76]. Minor editorial modifications were done to the content of [76]. For example, given the double-blind nature of its review, the original manuscript referred to our own previous work in [184] as "Marques *et al.*"; in this chapter we specify that [184] was another contribution of this thesis.

3.1 Introduction

This work focuses on the detection of pelagic species in acoustic backscatter data. Multi-frequency echosounders enable the acquisition of time series of acoustic backscatters during underwater acoustic surveys. Echosounder data can be visualized in 2D images known as echograms, where the vertical axis shows the depth or range in the water column, and the horizontal axis represents the time (see Figure 3.1). The intensity of each pixel corresponds to the reflected echo amplitude or intensity, generally computed as the volumetric backscatter strength. Different species, geological formations, and various phenomena (*e.g.*, suspended sediments) produce different echoes. Thus, an echogram will display a variety of structures and patterns that may be indicative of the seabed, the water-air interface, and the presence of one or many biological objects in the water column, usually schools of fish and/or planktonic organisms [185].

Underwater acoustic surveys allow marine biologists to gather large quantities of data that enable them to perform a variety of tasks crucial for Environmental Monitoring, such as species identification, biodiversity mapping, and animal behaviour studies, in a non-invasive manner. Detecting pelagic species (such as schools of herring and juvenile salmon) over large periods of time constitutes an important part of fisheries and ocean resources management, as well as valuable information towards a better understanding of the effects of climate change on the oceans. Echograms are commonly interpreted with manual or semi-automatic methods, using commercial software like Echoview[1]. Given the shear size of the data to analyze, this is a time-consuming process, prone to errors and inter-expert disagreements. Indeed, echogram analysis can be challenging due to many factors, including the varying size and acoustic properties of the targets, significant inter-class similarities, and the specific context of the data acquisition. For instance, a marine biologist will determine a type of fish based on location, time, behaviour, acoustic backscatter strength, differences in acoustic backscatter strength from different frequencies, and additional data from net tows and/or underwater cameras [184].

In this paper, we propose to detect pelagic species from echograms using a Deep Learning (DL) framework based on *instance segmentation*. More specifically, we aim to detect schools of herring and of juvenile salmon, which can be found concurrently in the same geographic locations (*i.e.*, within the same echograms), but typically at

[1]https://www.echoview.com/

different depths. In addition to the more general challenges of echogram analysis mentioned above, from a Computer Vision viewpoint, challenges related to the identification of pelagic species include the potential close proximity of different schools, making it harder to detect and distinguish between them; the possible close proximity of schools of juvenile salmon to the surface, which may then overlap with the turbulence of the water-air interface; the potential presence of bubbles around a school, altering the apparent morphology of the school; the possible small size of schools of juvenile salmon compared to the size of the echograms, which may make the feature extraction process for identification purposes less reliable. There are several available image analysis paradigms for the identification of pelagic species from echograms: image classification, semantic segmentation, object detection, and instance segmentation. We elect to use an instance segmentation paradigm here, which assigns an instance label locally to each pixel in the echogram, tackling precisely (at the pixel level) the *what* and *where* and distinguishing between different instances of the same class, in our case different schools of the same species. This allow us to address many of the aforementioned challenges related to overlaps and morphology, in addition to allowing for more precise biological analyses.

Our contributions are two-fold. 1) From a methodological viewpoint, we provide a comprehensive experimental design considering diverse feature extraction backbones within an instance segmentation framework adapted for echograms, taking advantage of the most powerful state-of-the-art DL architectures. 2) From a practical viewpoint, we show that instance segmentation networks are more suitable and accurate for the detection of pelagic species than object detection networks; the proposed instance segmentation framework offers a unique opportunity for automatic biological analyses based on a precise identification, at the pixel level, of schools of herring and of juvenile salmon. To the best of our knowledge, the proposed framework is the first of its kind in fisheries and acoustics (see Section 3.2). It is capable of detecting pelagic schools with an accuracy that closely matches that of human operators (see Section 3.4). Differently from what an expert can do manually, it also specifies the pixels associated with each detection allowing for a precise estimation of the number of specimens per school (see sub-section 3.4.6). Other advantages over object detection networks include the ability to better distinguish between schools in close proximity and the additional information on the often complex morphology of each output at the pixel level, instead of simple bounding boxes. Our method can be easily reproduced in other layouts/datasets and is also inherently scalable due to the use of a DL-based

instance segmentation network: it can identify new species as long as training pixel-level annotations are provided.

The remainder of the paper is divided as follows. Section 3.2 reviews related works on marine species detection in echograms. Section 3.3 details the Pixel-level Herring and Salmon (PLHS) dataset and our proposed instance segmentation framework. Section 3.4 discusses experimental results, including a comparison with the object detection framework we first proposed in [184]. Section 3.5 presents concluding remarks.

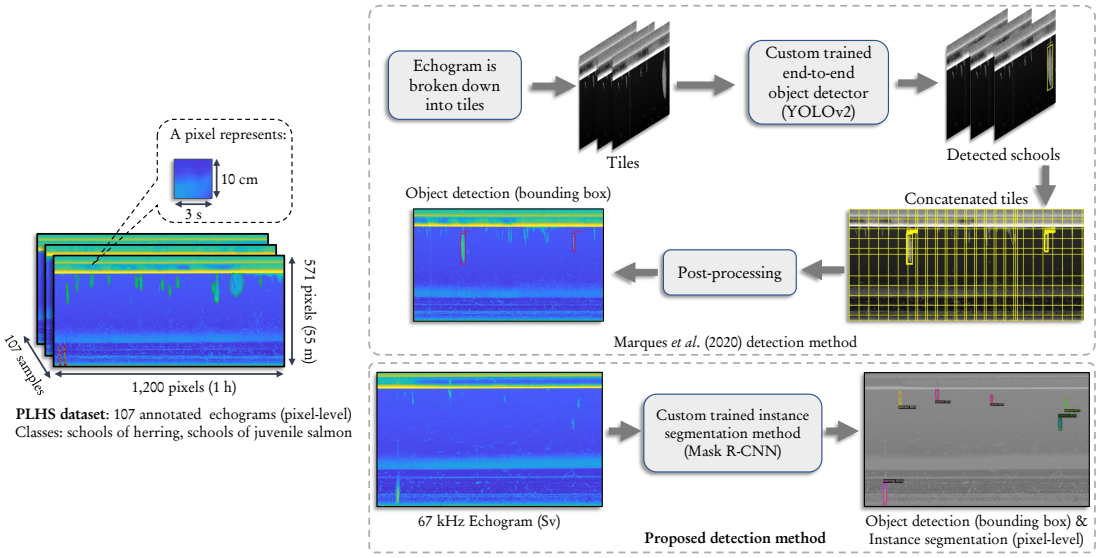


Figure 3.1: Our marine species detection methods proposed in this work and in [184] (serving here as a comparison baseline). PLHS (left) is a 107-sample dataset where schools of herring and of juvenile salmon are annotated on a pixel level. Using PLHS, we train the proposed, fully-custom instance segmentation-based detector (bottom right) as well as the method we first introduced in [184] (top right).

3.2 Related Works

The detection of pelagic species, and more generally of marine species from echograms can be categorized into classical Machine Learning (ML)- and DL-based approaches. Both categories are reviewed next.

3.2.1 Classical Machine Learning-based Approaches

Classical approaches to echogram analysis make use of hand-crafted features focused on statistical characteristics of organism aggregations. We find three different groups

of characteristics in the literature [186, 187, 188]: 1) positional/bathymetric characteristics, which relate to the position in the water column; 2) morphometric characteristics, linked to the school height, width, and perimeter; and 3) energetic characteristics, pertaining to the backscattered signal properties. Hand-crafted features are typically extracted using commercial software tools (*e.g.*, Echoview), based on the above domain-dependent taxonomy [184].

Hand-crafted feature-based methods for marine species detection in echograms generally rely on classical ML methods for feature classification. To detect various fish schools (Bonaerensis anchovy, Patagonian anchovy, rough scad, sprat, longtail hoki, and blue whiting), Cabreira *et al.* [82] compared three types of Artificial Neural Networks (ANN) architectures and found that for asymmetrical numbers of input data per species, the best ANN differed from one species to another, while for symmetrical data, Self-organizing Maps (SOM) yielded the best performance. This work led to ECOPAMPA [83], a recent tool for automatic fish schools detection and assessment from echo data based on the same ANN architectures. Also comparing different types of ANNs and Support Vector Machines (SVM), Robotham *et al.* [84] classified schools of anchovy, common sardine, and jack mackerel. They found that Multilayer Perceptrons (MLP) and SVMs performed better for multi-class classifications. Working with high-resolution echograms, LeFeuvre *et al.* [189] detected Atlantic cod and capelin using a Mahalanobis distance classifier. Also using Mahalanobis distance information, Charef *et al.* [190] identified three broad fish groups using a discriminant function analysis. Focusing on the classification of six mesopelagic fish groups, Gauthier *et al.* [191] proposed a decision model based on an objective classification decision tree. Fallon *et al.* [85] favored Random Forests (RF) to classify Southern Ocean krill and icefish echoes. More recently, Proud *et al.* [192] also proposed to use RFs to detect schools of silver cyprinid from echograms for generating consistent biomass time series. Moreover, RFs have also been used by Mannocci *et al.* [193] in the context of tropical tuna purse seine fisheries. The authors trained RFs to differentiate between high and low bycatch occurrence in data collected by echosounder buoys attached to drifting fish aggregating devices.

An important drawback of hand-crafted feature-based methods is that new sets of features need to be engineered for each species, making the methods not easily and readily scalable to various and diverse marine species. This drawback is mitigated by DL-based approaches such as the one we propose.

3.2.2 Deep Learning-based Approaches

DL-based approaches have been shown to achieve excellent results for a variety of Computer Vision-related applications in the visible spectrum, including for object detection. To this date, there is still only a handful of works utilizing DL beyond the visible spectrum for the detection of marine species from acoustic backscatter data. We direct the interested reader towards the survey work of Malde *et al.* [194] for a more general overview on the data-driven and DL-based future of marine science.

In a hybrid study involving both classical ML and DL methods, Shang and Li [195] studied echo features and classification methods of fish using simulated data. They experimented with three different features (all based on backscatters, *i.e.*, echo waveforms, echo spectrograms, echo spectra) and four different classification methods (decision trees, adaboost [196], ANNs, and Convolutional Neural Networks (CNN), and found that the best combination was composed by CNNs with echo spectrograms. Hirama *et al.* [86] detected five fish species (yellowtail, salmon, squid, sardine, and juvenile tuna) from echosounder data in a set-net using a CNN. With this image classification-based approach, the echograms have to be divided in a rough set of anisotropic non-overlapping tiles classified individually, assuming only one class of fish per tile. In a slightly different direction more in line with Natural Language Processing (NLP), Måløy [197] focused on the spatiotemporal properties contained in echograms. The author proposed a transformer-based approach that interprets the spatiotemporal dynamics of echograms through attention mechanisms to classify fish behavior and detect the onset of pancreas disease in farmed Atlantic salmon. Closer to our work, Brautaset *et al.* [87] focused on acoustic classification in multifrequency echosounder data for sandeel detection. They proposed a semantic segmentation CNN based on the U-Net [119] architecture. Their method yielded a substantially higher performance compared to that of Korneliussen *et al.* [198], who used a *traditional automated processing pipeline* to detect sandeel. In [199] we proposed a hybrid approach to detect schools of herring from echograms, in which Region of Interest (ROI) are first extracted based on schools' intensity and morphology and then classified via a DL-based classifier. We compared three popular CNN architectures for the feature extraction and classification task and found that DenseNet [200] achieved the best overall performance. One drawback is that the ROI extraction is species-specific and cannot be straightforwardly extended to other species. In a follow-up paper *et al.* [184], we provided a comparative study covering the entire spectrum of learning

approaches, from traditional and hybrid methods to complete end-to-end DL object detection networks. Focusing on schools of herring, we concluded that the latter are preferable to other learning approaches, providing comparable or better results than traditional methods even with limited training data (see sub-section 3.4.1 for details).

A limited number of papers focus on DL methods for the detection of marine species from sonar data, which generally have a higher resolution compared to echograms. They tackle the detection of jellyfish [201] and fish count/concentration [202, 203]. Neupane and Seok [204] provide a review of DL-based approaches for the more general topic of automatic target recognition from sonar data.

Deep Learning approaches are becoming more popular in fisheries acoustics, but instance segmentation is notably absent: recent works [87, 184] focused on the adjacent approaches of semantic segmentation and object detection, respectively.

3.3 Proposed Approach

Our proposed pelagic schools detector is composed of a custom, DL-based instance segmentation framework that was trained using PLHS, a novel proprietary dataset of schools of herring and juvenile salmon (see Figure 3.1 and sub-section 3.3.1). The method we debuted in [184] closely relates to the task the proposed system attempts to perform, thus it is used as the main baseline of comparison (see sub-section 3.4.1 for details). Figure 3.1 gives an overview of our framework and compares it to the framework we previously proposed in [184]. As seen on the bottom right of this Figure, our proposed method is able to: 1) identify schools of juvenile salmon and of herring; 2) specify the pixels composing each instance; 3) provide bounding boxes around each detection.

3.3.1 PLHS dataset

The Pixel-level Herring and Salmon (PLHS) dataset consists of 107 echograms with pixel-level annotations indicating the presence of schools of herring and of juvenile salmon. It contains 153 instances of schools of herring and 252 instances of schools of juvenile salmon, covering schools with different morphologies, positions in the water column, and biological densities. The annotations follow the MS-COCO format [205]. Particularities of this dataset include the fact that it can be used to train models aiming to perform multi-class object detection, but also semantic or instance segmen-

tation tasks, due to the granularity of its annotations. It also identifies two distinct pelagic species that are often difficult to differentiate even by specialists, resulting in an equally challenging automatic detection task. Figure 3.2 shows sample annotated echograms in PLHS.

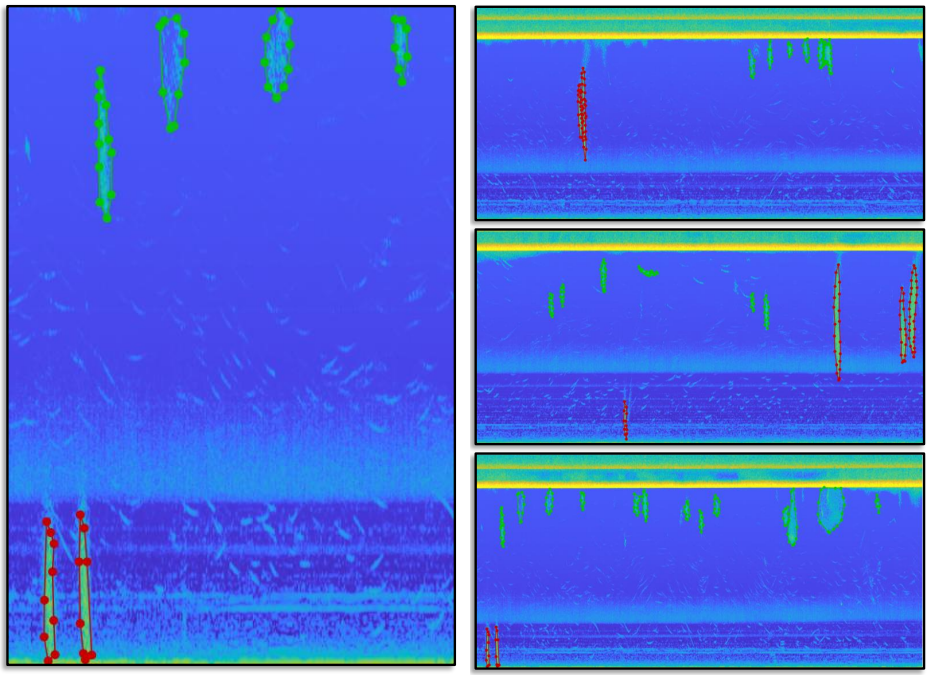


Figure 3.2: The Pixel-level Herring and Salmon (PLHS) dataset. **Left**: a zoomed-in region showing annotations of schools of herring (red) and juvenile salmon (green). **Right**: three samples illustrating the diverse nature of the dataset: most samples include multiple instances of schools of herring and salmon.

The acoustic data used to create the echograms in PLHS were obtained from Canada's Department of Fisheries and Oceans (DFO) and acquired using AZFP echosounder instruments [129]. These AZFPs were deployed by DFO in fixed positions close to the sea bottom looking upward under water columns of approximately 55 m. The instruments were located at the Okisollo channel, off the coast of British Columbia, Canada, between the months of May and October of 2015 and 2016. Each AZFP measurement is done at four frequencies: 67, 125, 200, and 455 kHz. The measurements at each frequency are visualized as a 571×1200-pixel echogram (water column depth x time) that represents one hour. Thus each pixel of a PLHS sample represents roughly 3 s throughout a depth resolution of 10 cm.

Since the acoustic response of schools of herring and of juvenile salmon is expected to be more pronounced at lower frequencies, we only consider the 67 kHz channel from

the multifrequency data that AZFPs capture. We create standard echograms that display volume backscattering strength (S_v). The S_v representation of acoustic data, often used to detect the presence of marine species, reflects the sum of all the acoustic response within a volume scaled to 1m^3. Given raw acoustic data from AZFPs, the S_v representation can be calculated as follows [129]:

$$S_v = EL_{max} - \frac{2.5}{a} + \frac{N}{26216a} - SL + 20\log R + 2\alpha R - 10\log(\frac{c\tau\Psi}{2}) \tag{3.1}$$

where EL_{max} represents the acoustic input (in dB re 1μPa) that the transducer has to receive to produce a full-scale output on the 16-bit A/D converter, a the slope of the detector response in units of volts/dB, N the number of "counts" (raw value) obtained from the instrument, SL the source level (dB re 1μPa at 1m), R the range of the instrument (m), α the absorption coefficient (dB/m), c the speed of sound (m/s), τ the transmit pulse length (s), and Ψ the equivalent solid angle that the transducer beam creates. The specific values of these parameters are available as metadata associated with each AZFP deployment. The AZFP instruments are calibrated by the manufacturer before each deployment.

Before carrying out the manual annotation process, we consulted with specialists from DFO, who provided important biological cues, such as: 1) schools of herring typically appear as elongated shapes in the vertical axis with a strong acoustic echo in the center of the school; 2) there are particular periods of the year (August-September) when the frequency with which schools of juvenile salmon are detected is expected to be reduced significantly; 3) schools of juvenile salmon often present themselves as smaller morphological structures than those representing schools of herring in echograms; 4) schools of herring will not be typically found travelling in close proximity to those of juvenile salmon; 5) schools of salmon usually travel closer to the surface. It is important to note that these biological cues might not necessarily be valid in other geographical regions. Figure 3.2 (left) illustrates a scenario where these cues were paramount to the annotation process: since the two schools closer to the sea bottom are easily identified as schools of herring, the smaller, sparser schools located at the top of the image are likely from juvenile salmon (as reflected by the annotations). Figure 3.2 (right) shows three samples from the PLHS dataset.

3.3.2 Instance Segmentation-based Framework

The proposed detection system is based on Mask-RCNN [120], considered as one of the state-of-the-art methods for instance segmentation. The official pre-trained implementation of Mask R-CNN [2] is trained on a dataset of natural images (COCO [205]) that structurally differ from our visual targets. Therefore, we initially re-trained all parameters from the Mask-RCNN architecture using the PLHS dataset to assess its ability to identify pelagic species in echograms. The performance observed in these initial experiments was rather low, likely due to the small size of the PLHS dataset and the complexity of the Mask R-CNN architecture.

Convolutional Neural Networks are able to automatically extract meaningful visual features from images of diverse natures. The fully connected layers of CNNs combine these features into *templates* that are representative of the different classes from a given dataset (given a successful training process). The feature-extracting and template-creating capabilities obtained with the pre-trained version of Mask R-CNN using COCO proved to be extremely useful to our application. We use *transfer learning* on the official implementation of Mask R-CNN to take advantage of these capabilities and fine-tune the framework to fit the two classes (*i.e.*, *salmon* and *herring*) of the PLHS dataset. In particular, we freeze the updating of parameters of the first block of Mask R-CNN (*stem*) as well as its first residual block [206].

Our proposed system was trained using a number of *backbone* models. Each model employs a different strategy for the extraction of visual features and requires an exclusive training process. We experiment with nine different combinations of backbones: **1)** ResNet-101 [206] with a Learning Schedule (LS) of $3x$; **2)** ResNet-101 with Feature Pyramid Networks (FPN) [207] and LS = $3x$; **3)** ResNet-50 [206] with LS = $1x$; **4)** ResNet-50 with LS = $3x$; **5)** ResNet-50 with Deformable Convolutions (DC) [208] and LS = $1x$; **6)** ResNet-50 with DC and LS = $3x$; **7)** ResNet-50 with FPN and LS = $1x$; **8)** ResNet-50 with FPN and LS = $3x$; **9)** ResNeXt-101 [209] with FPN and LS = $3x$. The *learning schedule* refers to the number of times that the original dataset (COCO [205]) was visited during pre-training (epochs): LS of $1x$ equates to approximately 12 COCO epochs, and $3x$ to approximately 37 COCO epochs. Feature Pyramid Networks are a mechanism proposed by Lin *et al.* [207] to represent feature maps at different scales, ultimately allowing for the identification of targets with significantly distinct dimensions. Deformable Convolutions were introduced by Dai *et*

[2] https://github.com/facebookresearch/detectron2

al. [208] to help CNNs better adapt to possible geometric transformations of the visual targets.

3.4 Experimental Results and Discussion

This section presents experimental results obtained using the PLHS dataset, including details on a comparison with our previously proposed method [184], training considerations, quantitative and qualitative evaluations, class-specific performance analysis, and a reflection about the capabilities of the proposed system.

3.4.1 Comparison Baseline

We compare the performance of the proposed system on the PLHS dataset with that of our previously proposed detection framework [184]. The method we introduced in [184] also works with data obtained with an AZFP, considers schools of herring as visual targets, and outputs detections as bounding boxes. In order to accomplish that, we created a system that first breaks echograms into smaller *tiles* of a constant size. These tiles are then used as the input of custom-trained object detectors based on YOLOv2 [210]. The output of the system goes throughout an aggregation (to consider detections from different tiles), yielding the final output. We retrained the model proposed in [184] with the PLHS dataset to include the *school of juvenile salmon* class and allow for a direct comparison with the present work.

3.4.2 Training Considerations

In the training routine of both our method and the comparison baseline, we used the PLHS dataset with a division of 73% for training/validation and 27% for testing. All models (see sub-section 3.3.2) are trained using a single NVIDIA™ GeForce GTX 1660 Ti GPU. We used the same set of hyper-parameters for the training of the proposed method in all configurations: 300 iterations, 2 images per batch, base learning rate of 0.02 (this learning rate drops linearly during training), 256 ROIs per image, and Stochastic Gradient Descent (SGD) with 0.9 of momentum as an optimizer. This particular set of hyper-parameters does not necessarily yield an optimal performance for all backbones; some larger models would likely benefit from longer training and from considering additional images per batch in a more robust hardware setting.

3.4.3 Quantitative Evaluation

Table 3.1 presents the performance of the proposed method along with that of the comparison baseline [184] for the various configurations/models, in terms of mean Average Precision (mAP). Both methods are evaluated exclusively on the test set of PLHS. As the comparison baseline does not provide pixel-level detection, we also report the performance of our method for bounding boxes (*i.e.*, "Obj. Det." column) for a direct comparison. Bold font indicates the best results for each metric. For instance segmentation and an Intersection-over-Union (IoU) threshold of 0.5, the best backbone configuration of our method is #4, which includes Mask R-CNN [120] with a ResNet-50 [206] backbone, no FPN and LS = $3x$. Its performance is closely followed by that of configuration #2, which differs in terms of backbone model (ResNet-101) and in the usage of FPN. When looking at mAP for IoU thresholds $\in [0.5 : 0.05 : 0.95]$, the situation is reversed, with configuration #2 yielding the best performance followed by #4. The worst performances are linked with the use of deformable convolutions [208] (*i.e.*, #5 and #6), which would require longer training routines. A similar performance is observed when considering the mAP for object detection (*i.e.*, using bounding boxes). Our method outperforms the comparison baseline significantly for object detection, by approximately 35 points (configuration #2 and IoU=0.5) and 34 points (configuration #4 with IoU $\in [0.5 : 0.05 : 0.95]$); and by approximately 15 points and 9 points for the worst-performing configurations (*i.e.*, #6).

Aside from its superior detection performance and more granular output, our method also executes about 10x faster than that our previous approach of [184]. Our method processes each echogram in $\sim$ 0.4 s (with small variations for each configuration) versus $\sim$ 4 s per sample for the system we proposed in [184]. This difference is mainly due to the overlapping tiling strategy of [184], which requires a full YOLOv2 inference for each tile.

3.4.4 Qualitative Evaluation

Figure 3.3 shows representative detection results of the best-performing configurations of both the proposed method (*i.e.*, configuration #4) and that of the comparison baseline [184] (*i.e.*, configuration #11). Although the baseline's results are qualitatively excellent on a first analysis, upon a closer inspection we identified two reasons explaining its significantly lower performance metrics (see Table 3.1). First,

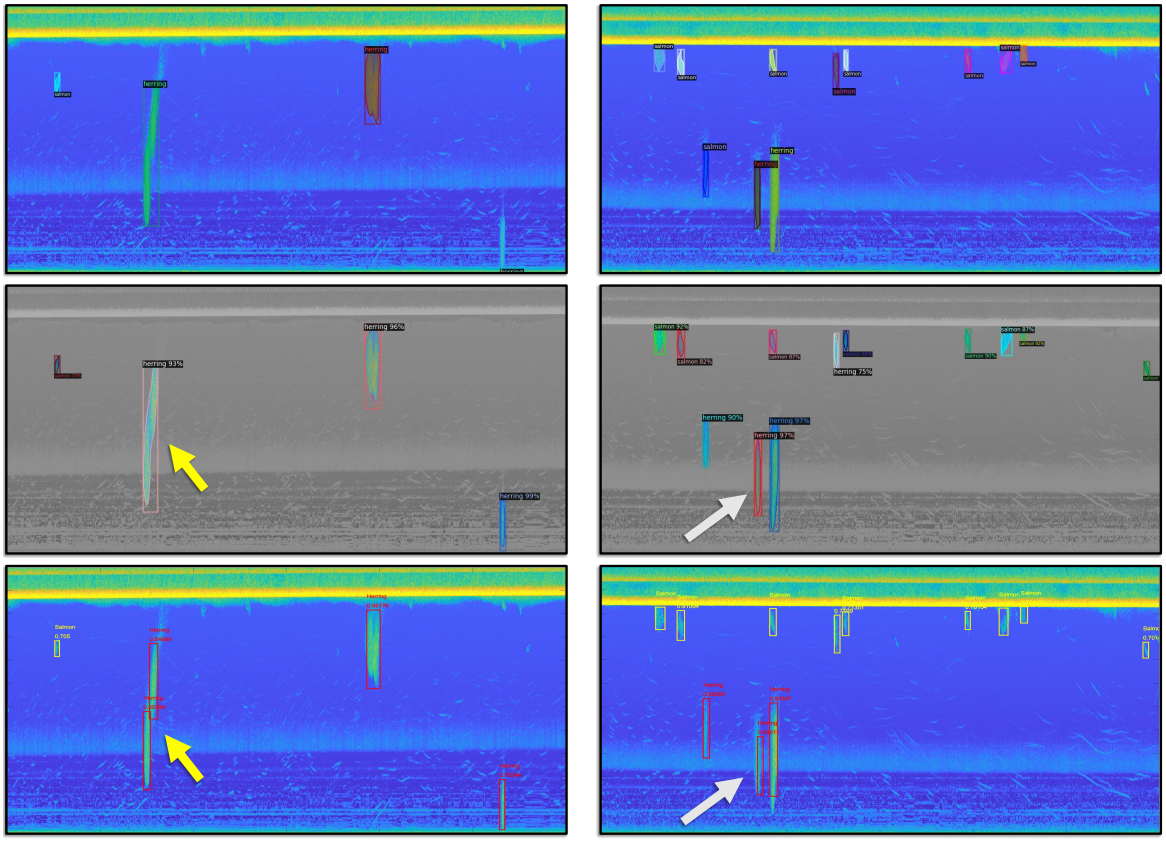
#	Configuration	Backbone	FPN[5]	LS[6]	Notes	mAP (Inst. Seg.) IoU=0.5	mAP (Inst. Seg.) IoU$_{95}$[†]	mAP (Obj. Det.) IoU=0.5	mAP (Obj. Det.) IoU$_{95}$[†]
1	Mask R-CNN [1]	ResNet-101[2]	N	3x		87.72	44.00	86.85	45.05
2	Mask R-CNN [1]	ResNet-101[2]	Y	3x		90.35	**52.79**	**90.15**	48.01
3	Mask R-CNN [1]	ResNet-50[2]	N	1x		89.63	46.08	89.63	47.76
4	Mask R-CNN [1]	ResNet-50[2]	N	3x		**92.12**	50.19	89.12	**50.48**
5	Mask R-CNN [1]	ResNet-50[2]	N	1x	Deformable convolutions [208]	73.8	29.61	70.79	26.99
6	Mask R-CNN [1]	ResNet-50[2]	N	3x	Deformable convolutions [208]	73.95	25.93	70.63	24.74
7	Mask R-CNN [1]	ResNet-50[2]	Y	1x		90.05	49.56	87.11	49.69
8	Mask R-CNN [1]	ResNet-50[2]	Y	3x		89.69	45.92	88.20	43.90
9	Mask R-CNN [1]	ResNeXt-50[4]	Y	3x	Aggregated residual transforms [209]	87.49	43.49	87.49	38.96
10	YOLOv2[3,7]	Darknet-53[3]	N	N/A	Tiling strategy [184]	N/A	N/A	41.69	11.63
11	YOLOv2[3,7]	ResNet-50[2]	N	N/A	Tiling strategy [184]	N/A	N/A	55.67	16.04

[1,2,3,4,5,6]: Mask R-CNN [120], ResNet-50 [206], YOLOv2 [210] and ResNeXt [209], Feature Pyramid Networks [207] and *learning schedule*, respectively.
[7]: A custom-trained version of the method we first proposed in [184].
[†]: mean Intersection-over-Union for thresholds in the [0.5 : 0.05 : 0.95] range.

Table 3.1: Mean Average Precision (mAP) comparison for the detection results on the test set of the PLHS dataset. Configurations 1-9 represent the instance segmentation-based method proposed, while the remaining layouts use the comparison baseline [184]. Best results are highlighted in bold.

the baseline often brakes a valid school down into two or more detections, creating false positives (see Figure 3.3a). Second, the baseline produces bounding boxes with a considerably worst fit with the ground truth than those generated by the proposed method (see Figure 3.3b). Despite these performance-lowering characteristics, we consider that our previous method generated mostly correct detections that carry significant scientific value in most of the test samples.

Despite our system's high performance (see Table 3.1), we consider that the metrics are still under-representing its actual capabilities. When qualitatively analyzing the predictions of our system, we notice a number of scenarios where detections of schools of salmon were triggered in regions not annotated as such in the dataset, but that closely resembled valid schools. Some of these "gray area" scenarios (as discussed in sub-section 3.3.1) could reasonably be considered as true positives, and would likely lead to different annotations if interpreted by different specialists. Figure 3.4 (top) illustrates this phenomenon: in this particular region, only three schools of juvenile salmon were annotated. The proposed method correctly identified these schools, but also indicated the presence of a fourth one (yellow arrow), which could have been considered as valid in the ground truth, based in part on the subjective analysis of the scientist annotating the dataset. Regardless, this "incorrect" detection hinders the performance of our system as reported in Table 3.1. A similar scenario is depicted



(a) A single school of herring is divided into two detections by the comparison baseline (yellow arrow).

(b) The bounding boxes created by our best-performing model (configuration #4 in Table 3.1) better fits the ground truth (gray arrow).

Figure 3.3: Qualitative comparison between the ground truth annotations (first row), best-performing configuration of the proposed method (second row) and best-performing version of our previous work [184] (third row).

in Figure 3.4 (bottom), where the two leftmost schools of herring are identified as four instances by the system. While this result could be interpreted as valid and is extremely useful for the timely analysis of echograms by scientists, these two extra detections are classified as false positives for performance evaluation purposes.

3.4.5 Class-specific Performance Analysis

We observed that schools of juvenile salmon are particularly challenging to annotate because their morphology and acoustic echo vary significantly across echograms. Conversely, schools of herring typically present easy-to-identify characteristics (*i.e.*, vertically-elongated shapes with strong intensities), leading to an overall easier anno-

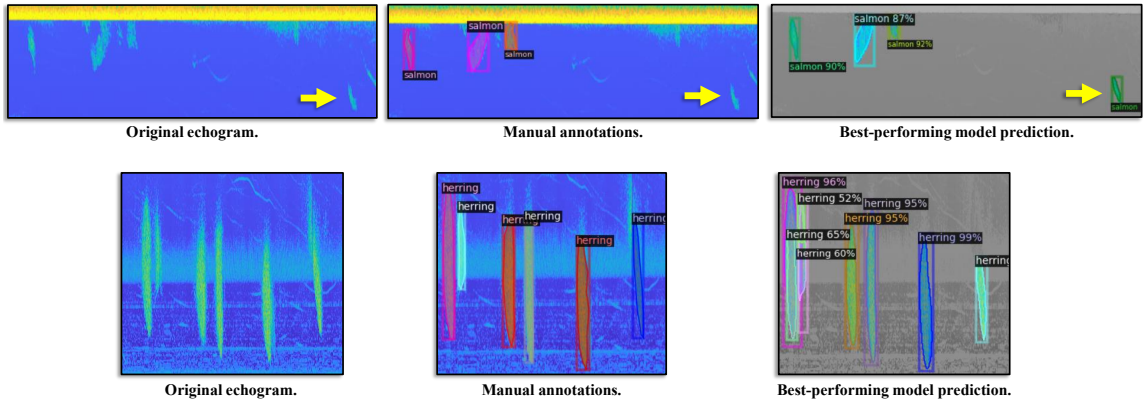


Figure 3.4: Two scenarios where the incorrect detections of the proposed system could be argued as valid. **Top:** The proposed system identifies possibly valid schools of salmon (yellow arrows) that are not annotated as such. **Bottom:** An instance where the proposed system broke two correct detections down into four distinct objects (leftmost schools).

tation process (see Figure 3.2 left). This phenomenon is echoed by the class-specific performances of our method. While Table 3.1 presents aggregate results that consider all classes of the dataset, we also compute the class-specific detection performance. Table 3.2 highlights the fact that the instance segmentation of schools of juvenile salmon is particularly difficult to perform, given that their morphology changes abruptly across samples. It also shows that the choice of configuration plays an important role on the system's capabilities. For instance, configuration #2 is preferable if the identification of schools of salmon is the focus of a study, while configuration #4 yields the best herring-specific performance.

		AP (Inst. Segm.)		AP (Object Det.)	
#	Configuration (see Table 3.1)	Herring	Salmon	Herring	Salmon
4	ResNet-50 $3x$	60.18	40.19	59.05	41.92
2	ResNet-101 FPN $3x$	58.29	47.3	47.61	48.42
7	ResNet-50 FPN $1x$	55.77	43.34	50.20	49.18

Table 3.2: Class-specific Average Precision (AP) for instance segmentation and object detection considering IoU $\in [0.5 : 0.05 : 0.95]$. Only the results for the three best-performing configurations detailed in Table 3.1 are presented.

3.4.6 Instance Segmentation vs. Object Detection

Instance segmentation methods are able to provide detailed information about their detection output; not only a list of pixels composing each detection is generated, but also a distinction between intra-class instances (*e.g.*, school of salmon A, school of salmon B). This ability allows for a precise estimation of populations associated with each detection, as illustrated in Figure 3.5, which is not possible via object detection. Consider, for example, that each pixel in a PLHS sample representing a school of herring contains approximately α specimens. The bounding box produced as the output of the object detection method in Figure 3.5 contains approximately $7,000$ pixels, while the manual annotation and instance segmentation outputs depicted in this same Figure have roughly $3,500$ and $4,200$ pixels, respectively. In this illustrative example, the instance segmentation output would result in a significantly better estimation of herring population (an error of 700α fish), while the bounding box-based object detection would have an estimation error of $3,500\alpha$ herring. The precise morphology of a detection, as offered by the proposed method, might carry vital information about schools of fish such as grouping and movement patterns, predation-related movements, environmental and anthropogenic stress, among others, which is not available via the bounding boxes of object detection methods.

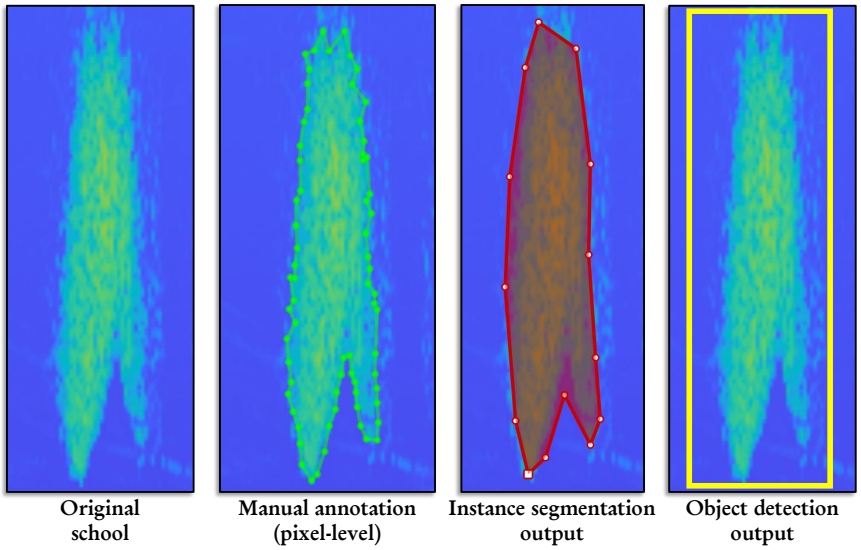


Figure 3.5: Illustration of different outputs and their influence on biological analyses. The precise morphology obtained with the output of instance segmentation methods allows for a better estimation of specimens count.

3.5 Conclusion

We propose a system that allows for a timely and precise identification of pelagic species (schools of herring and of juvenile salmon) from acoustic backscatter images (echograms). The proposed system uses a Deep Learning-based instance segmentation framework, the first of its kind in fisheries and acoustics, to generate not only bounding boxes around objects, but also to identify the groups of pixels that form each detection. This opens up many possibilities in terms of automatic biological analyses from underwater acoustic survey data. Our method comfortably outperforms the object detection framework we previously proposed in [184], while providing more information (*i.e.*, pixel-level data) as output in shorter processing times. The training and evaluation is done using PLHS, a novel dataset of pixel-level annotations of schools of herring and of juvenile salmon in echograms. We argue that the performance of our system is comparable to that of human operators, given that some of its incorrect detections could be considered as valid based on subjective decisions made during the annotation of PLHS (see sub-section 3.3.1). Future work on this system will involve a standardization module that allows for echograms coming from multiple instruments and deployment layouts to be used as input, as well as a semantic segmentation-based module dedicated to the identification of krill and hake (see Chapter 4).

3.6 Acknowledgments

The work described in this Chapter was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and ASL Environmental Sciences through the Alliance Grants program. Generous in-kind support in the form of active acoustic Environmental Monitoring data and biological expertise was provided by Canada's Department of Fisheries and Oceans.

Chapter 4

U-MSAA-Net: A Multi-Scale Additive Attention-based Network for Pixel-level Identification of Finfish and Krill in Echograms

As a natural progression of the work presented in Chapter 3, this Chapter details a novel system proposed for the identification, on a pixel level, of two different biological targets: schools of finfish (hake) and swarms of krill. The challenge posed by this new goal lies in the fact that these targets present themselves in echograms as morphological structures with notably heterogeneous, cloud-like shapes. Thus the use of Computer Vision-based solutions that assume the existence of *instances* (*e.g.*, instance segmentation) is not possible, prompting the use of semantic segmentation approaches that focus on pixel-level outputs.

We propose a novel DL-based semantic segmentation network named U-MSAA-Net, which introduces Multi-Scale Additive Attention (MSAA) modules. The novel, attention-based MSAA module allows the network to leverage contextual information from features maps of different scales towards the execution of an efficient semantic segmentation.

The performance of U-MSAA-Net is carefully analyzed in a comparative study involving semantic segmentation systems based on more "traditional" Machine Learning techniques, as well as in recent Deep Learning architectures. The experiments of this chapter are based on two datasets: FFK, a novel pixel-level dataset of echograms

containing krill and finfish, and on PLHS, which we introduced in [76].

Within the overarching theme of this thesis, the present work helps further illustrate that off-the-shelf and custom Computer Vision techniques can be employed for the efficient interpretation of Environmental Monitoring data, even in scenarios where they contain hard-to-distinguish marine species in the *underwater active acoustic* stream.

The content of this chapter was submitted for publication with negligible editorial modifications to the IEEE Journal of Oceanic Engineering and is currently undergoing peer-review. The author of this thesis, who is also the first author of the work from this chapter, was responsible for: conceptualizing, training and testing U-MSAA-Net; all experiments (with the exception of those involving the texture-based models); creation of the FFK dataset and leading the writing and revisions of this work. Moreover, the author would like to highlight the substantive contribution offered by the co-authors of this manuscript; in particular, Dr. Melissa Cote led the exploration and documentation of the texture-based methods described in sub-section 4.5.1, while Dr. Alireza Rezvanifar developed and documented the method used to calculate the signal-to-noise ratio (SNR) representations discussed in sub-section 4.4.2.

4.1 Introduction

In this work we aim to perform pixel-level detection of co-occurring marine species, finfish and krill, from underwater acoustic backscatter time series, utilizing a novel, deep learning (DL)-based computational pipeline.

4.1.1 Context

Underwater acoustic backscatter data are typically acquired at multiple frequencies during acoustic surveys via echosounders. Echosounders, such as the AZFP instrument [129], ping the underwater environment periodically with a conic shape-like signal in a given direction, reading back the acoustic responses (echo return) of objects present in the water column, biological or otherwise. Various species and phenomena (such as air bubbles) yield distinct acoustic responses at different frequencies. These responses are used to identify and locate objects of interest in the data. The data are visualized as echograms, *i.e.*, 2D images in which the vertical axis represents the depth within the water column, the horizontal axis represents time, and the pixel intensities encode the volumetric backscatter strength (denoted S_v).

The coastal waters of Vancouver Island have large zooplankton populations supported by a rich nutrient supply. This rich zooplankton community includes large Euphausiid populations(also known as krill) [211]. These krill are a primary food source for numerous finfish species, including hake and herring [212]. Krill are also harvested to provide food for aquaculture and aquariums [213]. Pacific Hake (*Merluccius productus*) are a migratory finfish species that typically travel from coastal California to the British Columbia coast to feed from June to October [214]. Euphausiids compose the primary prey for hake with one study finding that krill account for nearly 90% of their daily ration [212]. This reliance on krill is reflected in the spatial distributions of hake, which overlap almost perfectly with that of Euphausiids [214].

Hake are an important fishery species, used for human food and fish meal, and more hake are caught on the North American west coast than all other groundfish combined. Since 2008, Canadian and US fisheries have worked together in a collaborative effort to assess hake stocks, which influences both American and Canadian fisheries limits [215]. Stock assessments of krill use net-tows along with bioacoustics to determine biomass, which defines a biomass calibration coefficient to relate to echo return. To make accurate stock assessments of krill, it is necessary that one is able to accurately discern krill from finfish such as hake in acoustic echograms [216]. For the

purpose of this study, we refer to the larger group of "finfish" instead of "hake" to be factually accurate, as our data (see Section 4.4) contain an overwhelming majority of hake with a few instances of schools of other finfish.

From a computer vision standpoint, detecting co-occurring finfish (mostly hake) and krill from echograms poses a number of challenges. Figure 4.1 shows two sample multi-frequency one-hour echograms covering about 310 meters, in which hake and krill are present. The yellow line corresponds to the water-air interface; any pixel above that line is thus outside of the water and of no interest in this case. Finfish with swim bladders like hake tend to yield a strong signal at multiple frequencies, here 67 kHz and 125 kHz, whereas krill, as a larger zooplankton, tend to yield a Gaussian-like signal centered on one frequency, appearing more strongly here at 125 kHz. On one hand, finfish generally aggregate in schools, but hake typically do not form schools with well-defined specific shapes as we may observe for other types of finfish like herring; individual fish can also be found. On the other hand, krill occur in swarms that have a fuzzy and diffuse appearance that is more or less dense. These two types of organisms may also mix and overlap. In Figure 4.1a, the lighter pixels in the region within the black rounded rectangle in the 67 kHz echogram mostly represent schools of hake, while in the 125 kHz echogram, we can see similar patterns representing the schools of hake, mixed with a cloud-like structure of krill within the black rounded rectangle. At 125 kHz, hake typically appear as more intense (larger contrast with the blue background) than krill, but this is not always the case: in Figure 4.1b, the krill formation is so dense that its intensity surpasses that of hake at 125 kHz. Figure 4.1b also shows how small and sparse hake can be (67 kHz).

4.1.2 Contributions

The main contributions of this work are:

1. From a theoretical point of view, we propose a DL-based image semantic segmentation framework, U-MSAA-Net, that uses the popular U-Net [217] as its main architectural reference and includes novel Multi-Scale Additive Attention (MSAA) gates. This proposed framework uses attention mechanisms to allow for an efficient suppression of the feature responses from image regions with lesser semantic value, ultimately increasing segmentation performance. U-MSAA-Net also performs a process called *spatial consistency check*, which allows for the processing of input images of any size.

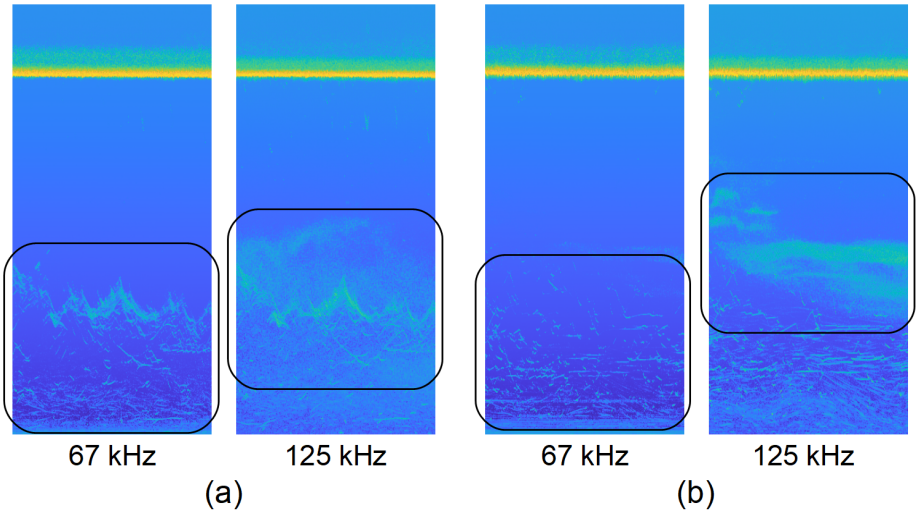


Figure 4.1: Some of the challenges of detecting co-occurring finfish (hake) and krill from multi-frequency echograms. Black rounded rectangles indicate regions in the 67 and 125 kHz echograms where hake and krill are present, respectively. In (a), the lighter pixels in the region within the black rounded rectangle at 67 kHz mostly represent schools of hake, while at 125 kHz, we can see similar patterns representing the schools of hake, mixed with a cloud-like structure of krill. At 125 kHz, hake typically appear as more intense (larger contrast with the blue background) than krill, but this is not always the case: in (b), the krill formation is so dense that its intensity surpasses that of hake at 125 kHz; the 67 kHz echogram also shows how small and sparse hake can be.

2. From a practical point of view, we apply U-MSAA-Net to the pixel-level classification of multi-frequency echograms to accommodate the detection of co-occurring species, finfish (hake) and krill, that have distinct and varied morphological signatures: the former tend to aggregate in small schools, while the latter tend to form cloud-like, diffuse structures. We also provide a comprehensive experimental comparison covering both ends of the learning spectrum, from traditional machine learning (ML) methods based on texture descriptors to other DL methods based on state-of-the-art semantic segmentation networks, which showcases the superiority of U-MSAA-Net.

The remainder of this paper is divided as follows. Section 4.2 reviews related works, looking at both traditional ML and DL methods for marine species detection in echograms. Section 4.3 describes our proposed methodology, starting with an overview of the architecture of U-MSAA-Net followed by details on the MSAA module. Section 4.4 presents our finfish and krill dataset and expands on its acquisition

and pre-processing for annotation purposes. Section 4.5 details the ML and other DL methods used in the comprehensive experimental comparison. Section 4.6 includes experimental results, training and implementation details, quantitative and qualitative comparative analyses, computational considerations, as well as the applicability of U-MSAA-Net to other data. Section 4.7 offers concluding remarks.

4.2 Related Works

Traditionally, echograms have been analyzed and interpreted by marine biologists using manual or semi-automatic methods facilitated by commercial software like Echoview [218] that allow users to define and extract various features from the data [77]. There are three broad categories of "handcrafted" features that are typically extracted, focused on statistical characteristics of aggregations of organisms [187, 188, 186]: 1) energetic (related to the backscatter signal properties), 2) bathymetric (related to the position in the water column), and 3) morphometric (related to the height, width, shape or perimeter of the aggregations). These features are then generally classified into species of interest using traditional ML methods. In recent years, we have seen a slow but increasing permeation of DL approaches to perform detection tasks in echograms. Here, we review both traditional ML and DL methods as they have been applied to echograms.

4.2.1 Traditional Machine Learning Methods

Traditional ML classifiers used in conjunction with handcrafted features extracted from echograms include decision trees and random forests (RFs), minimum distance classifiers, gradient boost classifiers (GBCs), support vector machines (SVMs), and various types of shallow artificial neural networks (ANNs).

Gauthier *et al.* [191] proposed a model based on an objective classification decision tree to distinguish between seven mesopelagic species assemblages (dominated by Euphausiids, pennant pearlside, and various lanternfish) corresponding to different acoustic mark types. Fallon *et al.* [85] classified Southern Ocean krill and mackerel icefish echoes via RFs, concluding that RFs could enhance the utility of incidentally collected acoustic data. Recently, Proud *et al.* [192] proposed RFs to detect schools of silver cyprinid for the purpose of generating consistent biomass time series, and Mannocci *et al.* [193] trained RFs to differentiate between high and low bycatch

occurrence in data collected by echosounder buoys in the context of tropical tuna purse seine fisheries.

LeFeuvre *et al.* [189] identified Atlantic cod and capelin from high resolution echograms with a Mahalanobis distance classifier. Charef *et al.* [190] also used Mahalanobis distance information to detect three broad fish groups (Japanese anchovy and round herring, jack mackerel and chub mackerel, lantern fish and pearlside) via a discriminant function analysis. They compared this approach to a Multi-layer Perceptrons (MLP) neural network and found similar classification rates. Minelli *et al.* [219] developed a semi-supervised ML system based on multibeam echosounder data that utilizes GBC (an ensemble classifier) to distinguish fish schools from other targets including gas bubbles and noise.

Comparing SVMs and two types of supervised ANNs (MLPs and Probabilistic Neural Networks (PNN)), Robotham *et al.* [84] classified schools of small pelagic species (anchovy, common sardine, and jack mackerel), and found that MLPs and SVMs outperformed PNNs in multi-class classification tasks. Cabreira *et al.* [82] compared three different ANNs (Self-organizing Maps (SOM), MLPs, and Radial Basis Networks (RBN)) for the detection of various fish schools including anchovy, rough scad, longtail hoki, sprat, and blue whiting, and found that the best performances were obtained by levelling the input data (number of schools) per species. For asymmetrical numbers of input data per species, the best ANNs changed with each species, while for symmetrical data, SOMs yielded the best performance. ECOPAMPA [83] is a tool for automatic fish schools detection and assessment from echo data based on those same ANN architectures.

Closer to our problem scope, Godínez-Pérez *et al.* [220] employed traditional ML and simple bi-frequency thresholding to remove micro- and macro-zooplankton echoes from echograms obtained from bottom trawls in order to highlight Pacific hake as the remaining signal.

4.2.2 Deep Learning Methods

DL methods that have been used in the literature to detect marine species from echograms are few to this day, although gaining more attention in recent years. They can be categorized according to the following four paradigms: image classification, object detection, instance segmentation, and semantic segmentation.

In an image classification-based approach, Hirama *et al.* [86] proposed to discrim-

inate between five fish species (yellowtail, salmon, squid, sardine, and juvenile tuna) using a Convolutional Neural Networks (CNN) architecture. They divided echograms into non-overlapping tiles then classified each tile, assuming only one type of fish per tile. Also focusing on image classification, our research group used a hybrid approach to detect schools of herring [199]. Regions of interest (ROI) in echograms were first extracted based on handcrafted features and then classified via one of three popular CNN architectures, with DenseNet [200] achieving the best overall performance. Given that this ROI extraction process was species-specific, the approach we proposed in [199] could not be directly extended to other species.

Our research group utilized object detection networks within a comparative study for detecting schools of herring that also looked at traditional and hybrid methods [77], and found that an end-to-end DL object detection framework, in particular YOLOv2 [134], was preferable to other learning approaches. YOLOv2 provided comparable or better performance than handcrafted features-based methods in a more timely manner, even when trained with only a small dataset. Another advantage of object detection networks was their ability to easily scale to new species (conditioned on the availability of representative and sufficient training samples).

Proposing an instance segmentation-based approach to detect pelagic species (schools of herring and of juvenile salmon), our research group showed in [76] (see Chapter 3) that the system outperformed the previous object detection-based methods [77], allowing for addressing challenges related to schools of varying sizes and placed in close proximity. The paper also argued that such a pixel-level detection method has the advantage of opening up many possibilities for automatic biological analyses, as it generates a precise identification of the pixel groups forming each detection. This dense detection also allows us to draw comparisons between our proposed U-MSAA-Net and the instance segmentation network of [76] using the Pixel-level Herring and Salmon (PLHS) dataset [76] (sub-section 4.6.5), in addition to our finfish and krill dataset (Section 4.4), to show the applicability of U-MSAA-Net to other data and contexts.

Closer to our proposed approach, Brautaset *et al.* [87] published a semantic segmentation network based on the U-Net [217] architecture to detect schools of sandeel. Their method yielded a substantially higher performance when compared to that of Korneliussen *et al.* [198], who used a non-ML "traditional automated processing pipeline" for sandeel detection.

More in line with Natural Language Processing (NLP), Måløy [197] proposed a

transformer-based approach to interpret the spatiotemporal dynamics of echograms through attention mechanisms, with the goal of classifying fish behaviour to detect the onset of pancreas disease in farmed Atlantic salmon.

4.2.3 Take-Aways

Traditional methods still play a major role in fisheries acoustics and are therefore of interest in comparative experiments such as the one we present in this paper (Section 4.5). Only a handful of works in the literature, covering the main image analysis paradigms, are dealing with the detection of marine species from echogram data via DL methods. Image classification, assigning a global label to an image, works in the presence of a single species only, unless additional mechanisms such as tiling or ROI extraction are put in place. Object detection, that is assigning a bounding box around each object of a given species in the echogram, yields an estimation of the "where" and "what" without precise information on the shape of the objects. Instance segmentation, assigning an instance label locally to each pixel in the echogram, tackles precisely (at the pixel level) the "where" and "what" and distinguishes between different instances of the same class. Both object detection and instance segmentation are suitable in the case of well-defined objects in space. Semantic segmentation, assigning a label locally to each pixel in the echogram, allows for the identification of which pixels belong to the background and which pixels belong to each of the species of interest, without any information on individual instances of objects. The characteristics of this paradigm make it suitable for the detection of amorphous aggregations such as those of krill, of central interest in this paper.

The two works that are closer to our own are that of Godínez-Pérez *et al.* [220], in terms of species of interest (planktonic/hake), and of Brautaset *et al.* [87], in terms of DL image analysis paradigm. Our work distinguishes itself from [220] and [87] by providing a novel multi-scale attention-based semantic segmentation network (Section 4.3), performing a thorough comparison with ML-based and DL-based methodologies (Section 4.5), and exploring the use of the proposed system in different contexts (sub-section 4.6.5).

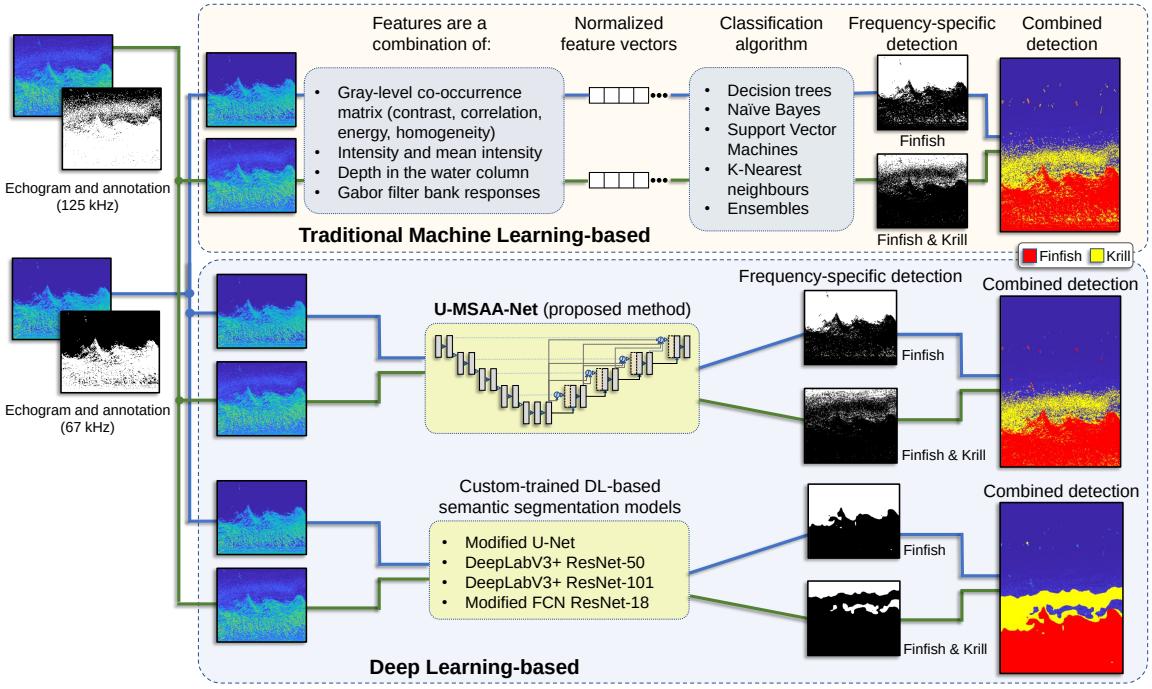


Figure 4.2: Flowchart of the proposed and compared methods for the pixel-level detection of finfish and krill. The proposed method (middle) is centered around U-MSAA-Net (Section 4.3), the compared traditional ML-based approaches (top) use different combinations of features and classifiers (sub-section 4.5.1), and the compared DL-based semantic segmentation approaches (bottom) include four different layouts combining models and backbones (sub-section 4.5.2). All approaches rely on multi-frequency echograms as input and yield frequency-specific outputs that are combined to extract krill (yellow) and finfish (red).

4.3 Proposed Approach

We propose a novel CNN-based architecture for the dense (*i.e.*, pixel-level) prediction of pelagic species in echograms named U-MSAA-Net. Each sample from our dataset (see Section 4.4) is composed of echograms reflecting measurements at two frequencies: 67 and 125 kHz. Given that 67 kHz echograms show mostly finfish while 125 kHz echograms represent a combination of finfish and krill (as explained in Section 4.1), the proposed approach is to use U-MSAA-Net to train semantic segmentation detectors specialized in each frequency. The 67 kHz detector performs a binary semantic segmentation task where pixels are classified as either foreground (*i.e.*, finfish) or background. Similarly, the 125 kHz detector performs a binary semantic segmentation task for which the foreground class stands for the co-occurrence of krill and finfish. Figure 4.2 shows the flowchart of the proposed approach (middle region), as well as

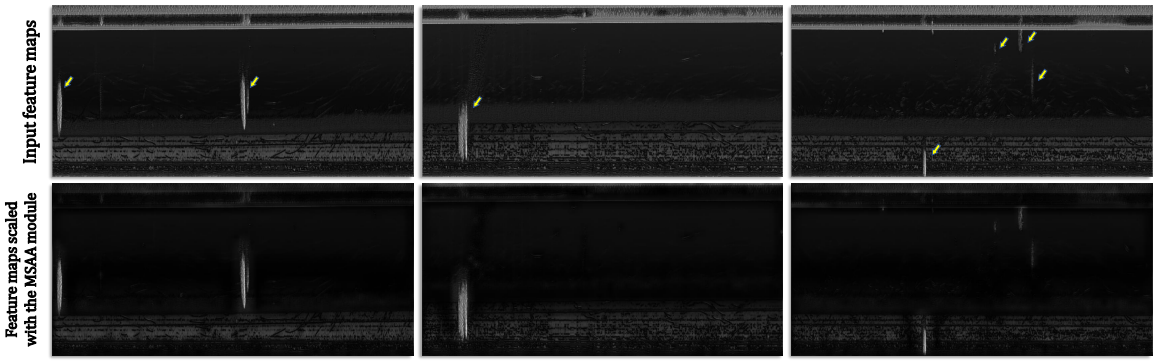


Figure 4.3: Effects of the proposed Multi-Scale Additive Attention (MSAA) module. The first row shows feature maps x^l used as inputs to the decoding phase of U-MSAA-Net (yellow arrows highlight target schools). After calculating attention map α^l (Equation 4.4) using MSAA, the proposed approach is able to determine $\hat{x}^l$, which places greater focus on regions of the echograms that carry relevant information to the segmentation task (second row).

that of the approaches (top and bottom) included in our experimental comparison. As illustrated in Figure 4.2, the final result of the proposed pipeline combines the output of both frequency-specific detectors trained with U-MSAA-Net: krill pixels are deduced as the spatial positions classified, simultaneously, as foreground at 125 kHz and background at 67 kHz. Finfish are assumed to be present in the remaining foreground pixels at 125 kHz. This also applies to the compared methods.

The main architectural reference of our network is U-Net [217]: a popular framework that offers excellent performance in segmentation tasks while maintaining a low usage of computational resources. U-Net achieves that by extracting visual features at multiple scales of an image, subjecting its inputs to multiple convolutional and pooling layers. This first phase of such semantic segmentation networks is often referred to as "encoding" (or contracting), as it progressively reduces the spatial dimensions and increases the semantic value of subsequent feature maps, in which each pixel represents a larger receptive field, ultimately assisting with the segmentation task. During the "decoding" (or expanding) phase, these networks both upsample intermediate feature maps closer to the original input dimensions and apply *skip* connections, *i.e.*, direct links between feature maps of corresponding spatial dimensions in both phases.

The progressively coarser feature maps obtained during the encoding phase are used to identify the overall location and class of larger aggregations in an image. The larger (*i.e.*, finer) features maps allow for more detailed and precise dense classifica-

tions; by combining coarse- and fine-level feature maps via skip connections, U-Nets harvest the capabilities offered by all levels of granularity.

Given that our visual targets (krill and finfish) present themselves in a combination of large sets of cloud-like connected components, small schools and even small individual specimen, U-MSAA-Net needs to allow for the efficient segmentation of both large- and small-scale objects alike. Oktay *et al.* [221] argue that this can be achieved with the use of Attention Gates (AG), modules capable of suppressing feature responses associated with background (or overall less-important regions) in feature maps. The goal of AGs is to determine attention coefficients $\alpha \in [0, 1]$ that scale each spatial location of a feature map (considering all of its channels), effectively forcing the network to focus on the salient regions that are more relevant to the segmentation task.

Since the AGs are employed during the decoding phase (see sub-section 4.3.1 and Figure 4.4), in our novel approach we feed them with all scales of coarse feature maps available at any given level of the upsampling stage of U-MSAA-Net. These multi-scale coarse feature maps serve as gating signals of our proposed Multi-Scale Additive Attention (MSAA) module, as detailed in sub-section 4.3.2 and Figure 4.5. Such gating signals allow for assigning lower weights (*i.e.*, smaller α) to regions of the images whose content is mostly background.

Figure 4.3 illustrates this process using the pixel-level herring and salmon [76] dataset (as its visual targets allow for an easy visualization of the effects of MSAA): the top row presents feature maps used as inputs to the skip connections of U-MSAA-Net (from level $l = 1$ of the decoding phase of Figure 4.4) before the MSAA-based scaling, with the yellow arrows indicating target schools. After considering coarser feature maps of multiple scales in the MSAA module to calculate α, U-MSAA-Net transforms the inputs into those shown in the second row of Figure 4.3. One should note that the MSAA-scaled feature maps place an almost exclusive focus on the regions where schools of herring and salmon are present. Conversely, the acoustic noise associated with the water-air interface observed in the upper region of the input is almost completely ignored. In the last column of Figure 4.3, the regions containing schools of salmon are also successfully highlighted (despite their reduced dimensions), while those representing background are efficiently suppressed.

An important positive side effect of using the MSAA module before skip connections is that less-relevant regions of the inputs are identified both during the forward and backward passes, ultimately optimizing the training process of the network's

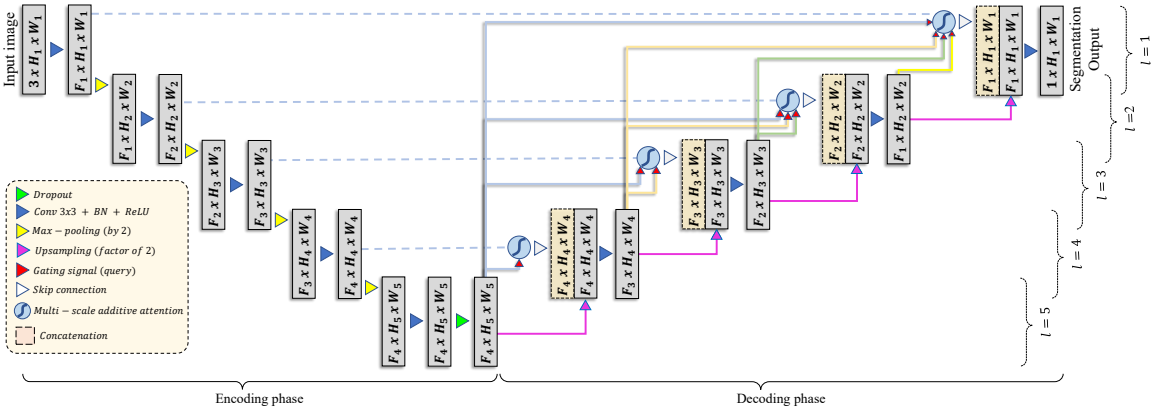


Figure 4.4: Architecture of U-MSAA-Net. During the encoding phase (left), a combination of convolutional blocks and max pooling layers are applied to reduce the spatial dimensions and increase the number of channels from feature maps. The decoding phase involves upsampling feature maps and concatenating them with inputs from the encoding phase that went through a scaling process based on the proposed MSAA module. The number of channels of U-MSAA-Net's output is dataset-specific.

parameters (as observed in [221]).

4.3.1 Architecture Overview

The architecture of the proposed semantic segmentation system is detailed in Figure 4.4. U-MSAA-Net is composed of five levels l in its *encoding* phase where the spatial dimensions of the input are progressively reduced while its number of channels and semantic value increase due to the growing receptive field of multiple convolutional layers. The channel-wise output of each level l is obtained via two sequential applications of the following convolutional block:

$$x_c^l = \sigma_r(\zeta(\sum_{c'=0}^{F_l} x_{c'}^{l-1} * k_{c',c})), \tag{4.1}$$

where x indicates a feature map, F_l and c represent, respectively, the channels in the input and output of a level l, σ_r denotes the rectified linear unit (ReLU) activation function ($\sigma_r(x) = max(x, 0)$), ζ and $*$ indicate, respectively, Batch Normalization (BN) [222] and convolution procedures (bias terms are omitted from Equations (4.1), (4.2) and (4.4) for notational clarity), and k stands for 3×3 convolutional kernels. Note that 1) the convolution operation is padded so that the spatial dimensions of the feature maps are only reduced with max pooling layers (yellow arrows in Figure 4.4),

2) one might have an intermediate number of output channels between the first and second applications of the convolutional block from Equation (4.1).

The convolutional block at $l = 5$ does not change the number of channels of the feature maps. Additionally, it is followed by a dropout layer [223] to mitigate the effects of overfitting. The *decoding* phase starts by upsampling, using bilinear interpolation, the feature maps at the end of $l = 5$. The progression to $l = 4$ happens by concatenating the upsampled feature maps of $l = 5$ with those of $l = 4$ from the encoding phase (skip connection). The concatenation is preceded by an MSAA-based scaling of the input from the enconding phase (as detailed in sub-section 4.3.2). This process repeats until $l = 1$ in the decoding phase; the final output of U-MSAA-Net has a number of channels that is determined by the number of classes and inputs from each sample of a dataset. For instance, when using the FinFish and Krill (FFK) dataset (see Section 4.4), the proposed solution involves the training of two individual U-MSAA-Nets with single-channel outputs (a binary segmentation result for both 67 kHz and 125 kHz echograms). When considering the single inputs (*i.e.*, 67 kHz echograms) of the PLHS dataset [76] (see sub-section 4.6.5), U-MSAA-Net is trained to produce two-channel outputs representing classification scores for the salmon and herring classes.

4.3.2 Multi-Scale Additive Attention (MSAA) Module

The goal of the MSAA module is to scale each pixel from the input feature maps of the skip connections (decoding phase of U-MSAA-Net) with attention coefficients $\alpha \in [0, 1]$. These input preserve higher spatial dimensions and, when combined with coarser feature maps from deeper levels of U-MSAA-Net, allow for a segmentation that observes representations that carry complementary semantic information (*e.g.*, coarser maps are typically associated with determining overall object location, whereas finer ones inform more precise, pixel-level decisions). Novel to the proposed MSAA module is that we use all coarser feature maps available at any given level of the decoding phase: for example, at level $l = 2$, U-MSAA-Net considers the feature maps obtained in levels $3, 4$ and 5 of this phase (see Figure 4.4 right). By leveraging multi-scale feature maps, the MSAA module yields an efficient suppression of feature responses from regions with lesser semantic value (*e.g.*, background-like) of the input of skip connections.

The multi-scale feature maps involved in the calculation of attention map α^l (*i.e.*, a

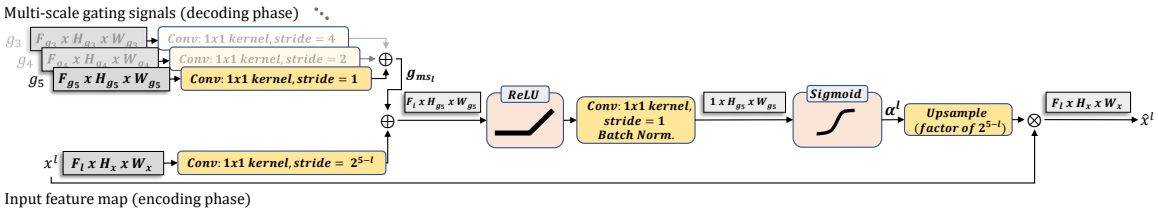


Figure 4.5: Proposed Multi-Scale Additive Attention (MSAA) module. Each pixel of input x^l is scaled using attention coefficients $\alpha \in [0, 1]$. This process is driven by spatial information collected from all multi-scale gating signals available at a decoding level l and a sequence of batch normalization [222] and activation functions. Attention map α^l is upsampled by a factor of 2^{5-l}, allowing for a pixel-wise multiplication with x^l. Output $\hat{x}^l$ is a feature map where semantically richer regions receive a greater focus.

single attention coefficient $\alpha \in [0, 1]$ for each spatial position of a decoding level l) are referred to as gating signals. The MSAA module uses additive attention [224], instead of multiplicative attention, because it has been shown to achieve higher accuracy despite its increased computational cost [221, 225]. The combined multi-scale-based gating signal g_{ms_l} employed by the MSAA module at a decoding level l is calculated as follows:

$$g_{ms_l} = \sum_{u=l+1}^{5} W_{g_u}^T g_u, \tag{4.2}$$

where l indicates a level of the decoding phase, g_u ($u \in \{1, 2, 3, 4\}$) denotes all gating signals represented by feature maps upsampled until a given level of the decoding, $W_{gu} \in \mathbb{R}^{F_{g_u} \times F_i}$ is a linear transformation computed using convolutions, while F_{g_u} stands for the number of channels of a gating signal g_u. The convolutions of W_{gu} are carried out with 1×1 kernels and are strided in such a way that their outputs have the same spatial dimensions as the coarser gating signal (g_5). Specifically, the stride s_{g_u} employed in the convolutions of linear transformation W_{gu} is given by:

$$s_{g_u} = 2^{(5-u)}. \tag{4.3}$$

The l-th level multi-scale gating signal g_{ms_l} allows for the calculation of attention map α^l as in Equation (4.4):

$$\alpha^l = \sigma_s(\zeta(\psi^T(\sigma_r(W_x^T x^l + g_{ms_l})))), \tag{4.4}$$

where x^l is the encoding input from level l (blue dotted lines in Figure 4.4), σ_s

represents a sigmoid activation function ($\sigma_s(x) = \frac{1}{1+e^{-x}}$), $W_x \in \mathbb{R}^{F_l \times F_i}$ and $\psi \in \mathbb{R}^{F_i \times 1}$ are linear transformations calculated using convolutions with 1×1 kernels, and F_l indicates the number of channels of x^l. To ensure that the spatial dimensions of x^l match those of g_{ms_l} (allowing for a pixel-wise addition), the convolutions of transformation W_x are strided by $2^{(5-l)}$. The resulting attention map α^l is upsampled by a factor of $2^{(5-l)}$ with bilinear interpolation, guaranteeing a match with the spatial dimensions of x^l. After a pixel-wise multiplication (considering all channels) between the upsampled α^l and x^l, U-MSAA-Net produces $\hat{x}^l$, which effectively focuses on regions with higher semantic value (as illustrated in Figure 4.3).

We use sigmoid σ_r as a normalizing activation function (instead of softmax) because it has been shown to assist with convergence during training [221]. Moreover, we empirically determined that the addition of a batch normalization ζ [222] after linear transformation ψ further helped with optimization and increased U-MSAA-Net's performance. The complete computational pipeline of our proposed MSAA module is illustrated in Figure 4.5.

4.4 FinFish and Krill (FFK) Dataset

We assembled a proprietary dataset, called FinFish and Krill (FFK), consisting of 200 one-hour multi-frequency echograms (67 and 125 kHz) with pixel-level annotations for each frequency indicating the presence of finfish and krill (present in all echograms), for a total of 77.4 million pixels. Reporting the total number of instances for each class is not applicable due to the species' morphological nature (see sub-section 4.2.3). The following subsections detail FFK's acquisition, annotation process, and a pre-processing strategy via a modified adaptive Wiener filter which allows for a more efficient annotation process.

4.4.1 Data Acquisition

The acoustic data were obtained from Canada's Department of Fisheries and Oceans (DFO). They were acquired with AZFP echosounders [129] deployed in fixed mooring positions close to the sea bottom looking upwards, with about 310 m of coverage in the water column, in the Brooks peninsula and Ucluelet sound off the coast of British Columbia, Canada, in 2017 and 2018. The echograms are 300×645 pixels (time $\times$ water column depth). Each pixel represents roughly 12 s and 0.5 m and shows the

S_v value, which reflects the sum of all acoustic responses within a volume scaled to $1m^3$. S_v values can be obtained from raw AZFP data as follows [129]:

$$S_v = EL_{max} - \frac{2.5}{a} + \frac{N}{26216a} - SL + 20\log R + 2\alpha R - 10\log(\frac{c\tau\Psi}{2}), \tag{4.5}$$

where EL_{max} is the acoustic input (in dB re 1μPa) that the transducer has to receive to produce a full-scale output on the 16-bit A/D converter, a is the slope of the detector response in units of volts/dB, N is the number of "counts" (raw value) obtained from the instrument, SL is the source level (dB re 1μPa at 1m), R is the range of the instrument (m), α is the absorption coefficient (dB/m), c is the speed of sound (m/s), τ is the transmit pulse length (s), and Ψ is the equivalent solid angle that the transducer beam creates. These parameter values are available as metadata that are associated with each deployment.

The echograms composing FFK offer a large and representative variety of samples, spanning nine months (September to December 2017, April to August 2018) and covering various hours of the day (between 6:00 am and 10:00 pm local time), which might affect species behaviour and consequently their apparent morphology.

4.4.2 Preprocessing for Annotation Purposes

Semantic segmentation systems, such as U-MSAA-Net, require pixel-level annotations of the FFK dataset for training and testing. Data annotation is a tedious and error-prone process for co-occurring schools of finfish and krill as the species are dense, granular and intertwined in the echograms (see Figure 4.1). To provide consistency and expedite this process, we designed a semi-automatic threshold-based hierarchical tool which can roughly segment the species at 67 and 125 kHz, in which the targets (*i.e.*, finfish and krill) are easily distinguishable. The user can then carefully fine-tune the results manually, representing a faster and more convenient option than annotating from scratch.

First, both finfish and krill are extracted as a single class at 125 kHz via a user-selected threshold (which typically varies from one echogram to another). Then, in a similar process, finfish are extracted at 67 kHz, due to the fact that most krill are typically not visible in these lower-frequency echograms (see Figure 4.1). The process produces two ground-truth (GT) masks: one showing finfish only (67 kHz) and one composed of finfish and krill together (125 kHz). Annotated pixels from the 125 kHz

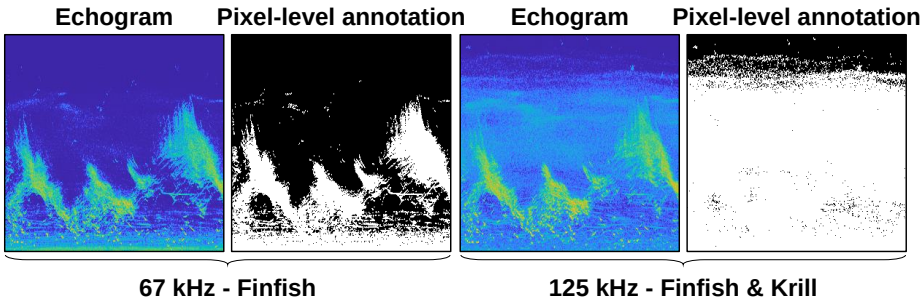


Figure 4.6: Sample pixel-level annotations for excerpts from a multi-frequency echogram: finfish only at 67 kHz (left), co-occurring finfish and krill at 125 kHz (right).

mask that are not also annotated in the corresponding spatial positions in the 67 kHz mask represent exclusively krill—the remaining ones indicating the presence of finfish. This hierarchical process allows us to annotate the dataset regardless of the spatial mix of the co-occurring species, as illustrated in Figure 4.6.

The amplitude of transmitted echoes is attenuated due to spherical spreading of the beam with range and absorption in water [226]. Echosounders compensate the depth-dependent attenuation of the received echoes with a time-varied-gain (TVG) function. Although this process makes target backscatter amplitudes consistent along the depth direction (*i.e.*, y-axis in echograms), it also amplifies background noise, hindering the thresholding step of our annotation process. To address this problem, we use the data quality improvement method of [227] with two proposed modifications (see "Mod. 1" and "Mod. 2" below) to compute SNR images and then use them for annotation purposes only (not used during training/testing by U-MSAA-Net nor by the compared ML and DL methods of Section 4.5). Thresholding is carried out on SNR images, in which each pixel value gives the relative amount of signal to background noise for that pixel, instead of S_v. Using the same notation as in [226], TVG can be removed from the measured volume backscatter, $S_{v,meas}$:

$$Power_{cal}(i,j) = S_{v,meas} - 20\ log_{10}\ r_{tvg}(i,j) + 2\alpha\ r_{tvg}(i,j), \tag{4.6}$$

where the index (i,j) represents the $i-$th ping and $j-$th depth sample, $Power_{cal}$ is the logarithmic measured power in the receiver after removing TVG, r_{tvg} is the depth value and α (dB m^{-1}) is the absorption coefficient. To estimate the background noise, it is assumed that the noise power does not change in each ping. We changed the way noise is calculated in [227] ("Mod. 1") for each water column as follows:

$$Noise(i) = min_{(i,j\in[1,\mathcal{N}])}[mean_{(k,s)\in\epsilon}(Power_{cln}(k,s))]. \tag{4.7}$$

The original equation in [227] does not give an accurate estimation if in some areas the *min* value of the sliding window returns a high number due to large schools or severe interference. In Equation 4.7, ϵ is a neighborhood around (k,s), $\mathcal{N}$ is the number of depth samples and $Power_{cln}$ is the echogram denoised with an adaptive Wiener filter:

$$Power_{cln}(i,j) = [\mu(i,j) + \frac{\sigma^2(i,j)}{\sigma^2(i,j)+v^2}(Power(i,j) - \mu(i,j)]_X. \tag{4.8}$$

We also changed the Wiener coefficient in Equation 4.8 from $\frac{\sigma^2(i,j)+v^2}{\sigma^2(i,j)}$ in [227] to $\frac{\sigma^2(i,j)}{\sigma^2(i,j)+v^2}$ ("Mod. 2"). In Equation 4.8, $Power(i,j) = 10^{Power_{cal}(i,j)/10}$, $\mu(i,j)$ and $\sigma^2(i,j)$ are the local average and local variance of $Power(i,j)$ in a $n \times n$ window around (i,j), respectively, v^2 is its global variance and X the number of times that the filter is applied. We set $n = 3$ and $X = 2$. After estimating $Noise(i)$ for each ping, TVG is added back to obtain the noise estimation of the entire echogram, *i.e.*, $S_{v,noise}(i,j)$:

$$S_{v,noise}(i,j) = Noise(i) + 20\ log_{10}\ r_{tvg}(i,j) + 2\alpha\ r_{tvg}(i,j). \tag{4.9}$$

The SNR image is then obtained as follows:

$$SNR(i,j) = 10\ log_{10}(10^{\frac{S_{v,meas}(i,j)}{10}} - 10^{\frac{S_{v,noise}(i,j)}{10}}) - S_{v,noise}(i,j). \tag{4.10}$$

Not only does TVG not affect SNR values along the depth direction, the relative intensity of objects is still preserved with respect to their neighborhood. Figure 4.7 shows the difference between $S_{v,meas}$ and SNR in terms of consistency of values in the depth direction. In Figure 4.7a, there is a large difference between $S_{v,meas}$ values of nearby water columns at different depths, where there is only background noise, whereas SNR values in Figure 4.7b are almost the same (a desirable outcome). Figure 4.8 illustrates the difference between using $S_{v,meas}$ and SNR images in our annotation tool at 125 kHz (a similar behaviour is observed for 67 kHz). Applying the lowest possible threshold on $S_{v,meas}$ images that allows to extract finfish and krill also extracts the upper part of the echogram as foreground (left). This issue no longer exists for SNR images (right). The threshold is only applied to pixels below a user-

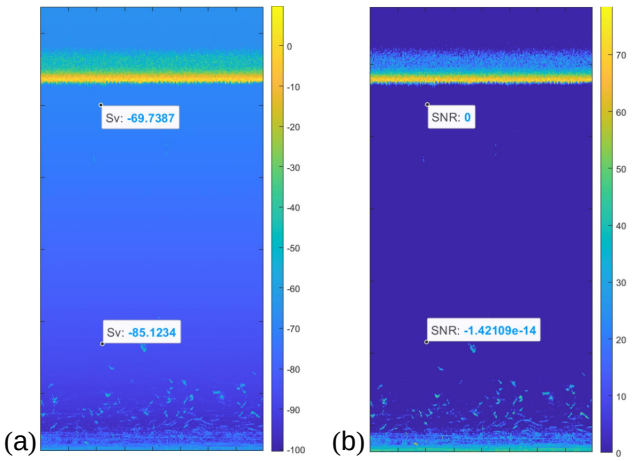


Figure 4.7: Sample $S_{v,meas}$ echogram (a) and its corresponding SNR version (b). Data tips show the respective values in nearby pings in areas where only background noise exists. Intensity values are in dB.

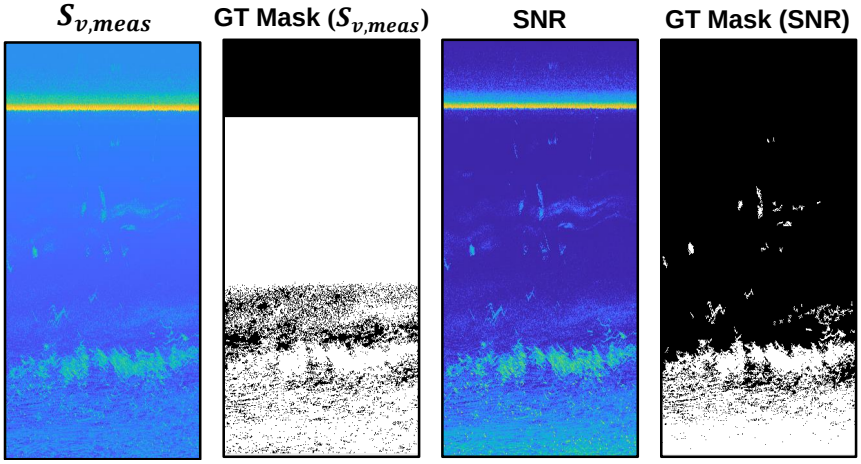


Figure 4.8: Differences in annotations using a single low threshold applied to $S_{v,meas}$ and SNR images at 125 kHz.

specified line (*i.e.*, a fixed y-axis value), with the pixels above this line omitted from the annotations.

4.5 Compared Methods

In order to perform a comprehensive experimental comparison of the proposed U-MSAA-Net for finfish and krill detection in echograms, we need to consider methods that cover both ends of the learning spectrum: traditional ML methods relying on handcrafted features and classifiers, as well as DL-based semantic segmentation net-

works. The rationale is that although the future of echogram analysis seems to be heading towards DL, traditional ML methods still play an important role in fisheries acoustics as seen in the literature review (Section 4.2). From that literature review, we can also infer that there has not been many published works dealing specifically with krill detection in echograms using machine learning. As a result, we cannot rely on existing published works for the purpose of our comprehensive experimental comparison. We thus provide here a detailed description of the compared methods that focus on textural cues for the traditional ML methods and on custom-trained versions of popular and varied semantic segmentation networks for DL methods.

The following sub-sections detail that each compared approach is implemented using a specific combination of features and models, as summarized in Figure 4.2 (top region for ML and bottom region for DL). All compared methods utilize the same strategy used with U-MSAA-Net, *i.e.*, a binary pixel-level classification of 67 kHz echograms for finfish detection and of 125 kHz echograms for a combined finfish and krill detection.

4.5.1 Traditional Machine Learning: Texture-based

Texture, as one of the important cues used by the human brain for interpreting images [228], has played an important role in image segmentation and classification tasks for many decades. While texture descriptors *per se* have somewhat taken a back seat to DL and its automatically extracted features in the last decade, we hypothesize that they remain an interesting and powerful source of information for describing the fuzzy, cloud-like diffuse appearance of krill within echograms. Texture descriptors are generally used in conjunction with traditional ML supervised learning techniques to perform whatever computer vision task is required, semantic segmentation in our case. They can be broadly categorized as statistical or spectral [229]. Statistical descriptors describe the spatial distribution of gray-level values within an image region, whereas spectral descriptors compute the response of an image region to a pre-defined filter bank. Considering the large number of available texture descriptors, our strategy was to include popular descriptors from each of the two categories: the Gray-level Co-occurrence Matrix (GLCM) [228] and Gabor filter banks [230]. Early experiments with Local Binary Patterns (LBP) [231] did not yield a good enough performance on our dataset to be included in the final experimental comparison.

The GLCM estimates second-order statistics properties of an image region, from

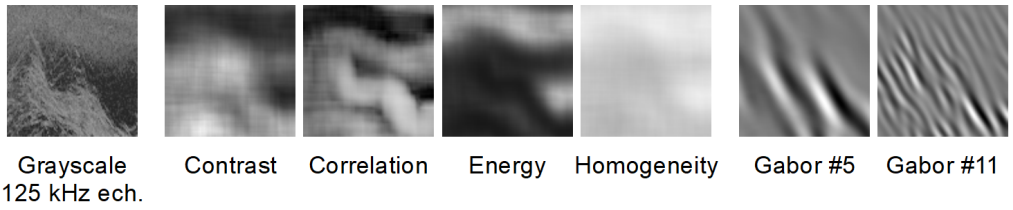


Figure 4.9: Sample GLCM features (middle) and Gabor filter responses (right) for a cropped 125 kHz echogram (left).

which we derive four standard features: contrast, correlation, energy, and homogeneity. To get features for each echogram pixel, we compute a GLCM for each pixel using a 25×25-pixel sliding window and considering 8 levels in the GLCM, yielding four GLCM features per pixel. We utilize the Gabor filter bank design proposed in [232], consisting of 24 filters covering six orientations at four scales, with lower and upper center frequencies of 0.05 and 0.4, operating on a 49×49-pixel window, yielding 24 Gabor responses per pixel. Figure 4.9 shows sample contrast, correlation, energy, and homogeneity images (middle), and sample responses of filters #5 and #11 (two scales at 120-degree orientation) (right) for a cropped region of a 125 kHz echogram. We combine the texture descriptors $(4 + 24)$ with the pixel intensity and mean intensity in the neighborhood (2) and depth information (1), to form up to a 31-dimensional feature vector. The rationale is that intensity information has proven very useful during the annotation process, and krill and finfish, although quite mobile throughout the day, remain within a given range of depths (*e.g.*, they cannot be found above the water). We experiment with variations on the above feature vector to form eight configurations (Table 4.1). Model #5 from Table 4.1 corresponds to the full feature vector. Features are normalized individually.

Table 4.1: Configurations for the compared traditional Machine Learning-based approaches.

#	Dimensions	Details
1	30	Intensity, mean int., GLCM features, Gabor responses
2	26	Intensity, mean int., Gabor responses
3	6	Intensity, mean int., GLCM features
4	2	Intensity, mean int.
5	31	Depth, intensity, mean int., GLCM features, Gabor resp.
6	27	Depth, intensity, mean int., Gabor responses
7	7	Depth, intensity, mean int., GLCM features
8	3	Depth, intensity, mean int.

Classification based on the feature vector can be done via various traditional ML classifiers. As we do not know beforehand which classifier will perform best, for each configuration in Table 4.1, we train five different types of classifiers: decision trees, naive Bayes, SVMs, K-Nearest Neighbors (KNN), and ensembles. Decision trees [233] predict classifications according to a tree-like structure, with a series of decisions based on individual predictors at each node. Naive Bayes classifiers (see [234]) estimate the parameters of a probability distribution assuming predictors are conditionally independent. At inference time, they classify test data according to the largest posterior probability. SVMs [235] map data as points in space such that points from two different classes are divided by a hyperplane that yields a maximal margin between the classes. At inference time, test data are labeled according to which side of the hyperplane they fall. KNN classifiers [236] do not need training *per se*; they predict the class membership of a test pixel on the fly according to a vote among the k closest "training" (reference) pixels in the feature space. Ensemble classifiers (see [237]) combine a set of trained weak learner models to obtain better predictive performance than could be obtained from any of the models alone. At inference time, labels are obtained by aggregating predictions from the weak learners.

4.5.2 Deep Learning: Other Semantic Segmentation Networks

Considering that there exists a large amount of successful DL-based semantic segmentation systems currently available [217, 238, 239, 240, 241, 242], our strategy for the experimental comparison was to implement techniques that occupy two opposite sides in the research spectrum: a now-classical, extremely influential semantic segmentation framework typically used as comparison baseline (Fully-convolutional Networks (FCN) [243]), and a technique often considered to be amongst the state-of-the-art in the field (DeepLabV3+ [244]). Additionally, we also explore the performance of the aforementioned U-Net (see Section 4.3) because of its relevance in the field and architectural similarities with U-MSAA-net.

FCNs [243] were proposed by Long *et al.* to create spatially dense predictions (*i.e.*, per-pixel classifications) using only convolutional layers (*i.e.*, without fully connected layers). The authors also proposed the use of skip connections that allow for the combination of coarser feature maps found in deeper layers with the finer characteristics offered by the feature maps of shallower layers. Given the results presented in [243], we expect the predictions of this model to be coarser, in contrast with the

finer output desired.

DeepLabV3+ [244] utilizes a combination of three important modules: 1) a common encoder-decoder structure where the spatial dimensions of an input are gradually reduced (while their semantic value increases), followed by an up-sampling path that recovers the original dimensions (similar to [243, 217]), 2) spatial pyramid pooling, with networks able to create features in multiple scales by using filter and pooling operations at different points of the architecture, and 3) atrous convolutions, which allow for the selection of arbitrary resolutions in the encoded features. Other aspects that contribute to DeepLabV3+'s efficiency are the use of batch normalization [222], ReLU and modifications to include recent feature extractors (Xception [245]).

Custom DL-based frameworks are commonly built using parameters that were previously trained on large datasets of natural images (*e.g.*, ImageNet [246], COCO [247]) via transfer learning. Our echograms, however, do not present natural-looking visual targets; for that reason, most of the DL-based layouts explored (#10, #12 and #13 in Table 4.2) were trained from scratch without the of use of any pre-determined parameters, allowing for a comparison with the ones that employed transfer learning (#9, #11).

4.6 Experimental Results

We compare the performance of the proposed system, U-MSAA-Net, with that of several state-of-the-art DL-based semantic segmentation methods as well as of traditional ML-based classifiers using handcrafted features (described in Section 4.5). The challenging task of pixel-level identification of krill and finfish offered by the FFK dataset (Section 4.4) is used for training and testing all methods. The remainder of this section first presents training and implementation considerations, followed by quantitative and qualitative evaluations on the FFK dataset, then computational considerations. The section ends with additional experiments on the PLHS dataset (pixel-level salmon and herring classes) [76] to further assess U-MSAA-Net's versatility and applicability to other data.

4.6.1 Training and Implementation Considerations

The FFK dataset is partitioned into 70%/10%/20% for training, validation, and testing, respectively, which amounts to 140/20/40 echograms. For the traditional ML

approaches, the validation and training sets are combined for a 80%/20% (160/40 echograms) partitioning.

DL-based Approaches

All models (U-MSAA-Net, U-Net [217], FCN [243], DeepLabV3+ [244]) were implemented using PyTorch [248]. Specifically, the U-Net architecture was implemented from scratch and slightly modified to allow for non-square inputs: in the original work [217], U-Net uses a tiling strategy to break odd-shaped inputs down into square patches. Instead, we perform a *spatial consistency* check that compares the spatial dimensions of feature maps during upsampling, and slightly resizes them with bilinear interpolation if necessary for matching (given that performing subsequent non-padded convolutions and upsamplings might create feature maps of slightly different spatial dimensions). The FCN networks were also implemented with spatial consistency checks, as well as with batch normalization [222] and ReLU in their convolutional blocks. For this reason, we refer to these versions of U-Net and FCN as "modified". The implementations of DeepLabV3+ utilized are those offered by PyTorch's torchvision [249].

The DL-based models were trained with a single NVIDIA™ GeForce GTX 1660 Ti GPU. The number of training epochs for each model is determined by the overall strategy (*i.e.*, if using transfer learning or not), model complexity, and level of stability from train and validation losses. Models #9 to #13 in Table 4.2 were trained for $20, 500, 100, 100$ and 500 epochs, respectively. The learning rate and batch sizes of DeepLabV3+ models (#9 and #10) were $1e^{-4}$ and 2, respectively, whereas the remaining layouts (#11 through #13) used $1e^{-3}$ and single-sample batches, respectively. All DL models were training with the ADAM [250] optimizer. Layouts #10, #12 and #13 from Table 4.2 do not use pre-trained weights. Models #9 and #11 are based on transfer learning of architectures pre-trained on the ImageNet [246] (#11) and COCO [247] (#9) datasets.

Given that semantic segmentation networks generate a continuous classification score for each pixel (ranging, across all models in our experiments, from approximately -0.2 to 1.2), one has to set a threshold above which pixels are considered to be from the *positive* class. For the FFK dataset, this class represents the co-occurrence of krill and finfish in 125 kHz echograms, or the occurrence of finfish in 67 kHz echograms. We designed a systematic approach that determines an appropriate

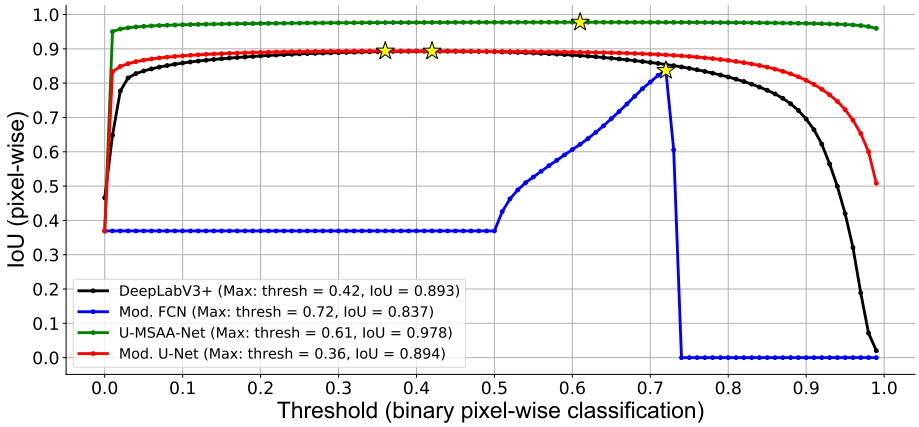


Figure 4.10: Hyper-parameter search on the validation set of FFK for the DL models #9, #11, #12 and #13 (see Table 4.2) at 125 kHz. Stars indicate thresholds that yield a maximum IoU for each model and that are thus used at inference time on the test set.

threshold by testing 100 different values in the $[0 : 0.01 : 1]$ range. For each threshold considered, all echograms from the validation set are used for inference purposes, and the Intersection-over-Union (IoU) considering all of their pixels is calculated. The model-specific threshold that yields the highest overall IoU in the validation set is used in the experimental results reported in Table 4.2. Figure 4.10 illustrates this hyper-parameter search process for four of the layouts reported in Table 4.2 (#9, #11, #12 and #13), *i.e.*, for each compared base architecture and for the proposed method. One should note that a suitable choice of threshold is highly dependent on the architecture considered, as well as the nature of the data being used to calculate the IoU.

Traditional ML-based Approaches

The compared ML-based models and features were implemented in MATLAB R2021a, except for the Gabor filter bank, which was implemented in C (code publicly available from the authors of [232]). Experiments were carried out on a PC with the following specifications: AMD Ryzen™ 7 5800H CPU, 16 GB RAM, and NVIDIA™ GeForce RTX 3060 GPU. For GLCM-derived features, experiments on hyper-parameter setting yielded the following optimal values: $L = 8$ and $W = 25$ pixels. The 160 available echograms for training amount to about 31 million training pixels for each frequency. Using all those training data would unnecessarily increase the complexity of the training process. Instead, we select a number of pixels T from each class of each training

image with uniform sampling, which allows us to cover the intra-class and inter-class variations well at a reduced complexity. Experiments have shown that $T = 200$ yields an optimal compromise between complexity and performance. We thus have used a total of $64,000$ training samples (160 echograms x 2 classes x T) for each frequency. For each feature vector configuration (see Table 4.1), we train several variations of the five types of classifiers presented in sub-section 4.5.1 using MATLAB's *Classification Learner* application, and keep the classifier that yields the highest validation accuracy using 5-fold cross-validation. The possible classifiers include coarse, medium and fine decision trees; Gaussian and kernel naive Bayes classifiers; linear, quadratic, cubic, coarse Gaussian, medium Gaussian, and fine Gaussian SVMs; coarse, medium, fine, cosine, cubic, and weighted KNNs; boosted tree, bagged tree, subspace discriminant, and subspace KNN ensembles.

4.6.2 Quantitative Comparison

Table 4.2: Semantic segmentation performance calculated echogram-wise for the proposed and compared approaches, with all reported values including the average and standard deviation on the test set of FFK. Bold font indicates the best-performing approaches of each category.

			IoU				Precision				Recall			
#	Configuration	Classifier/Backbone	125 kHz †		67 kHz ‡		125 kHz †		67 kHz ‡		125 kHz †		67 kHz ‡	
1	Int-GLCM-Gabor	Bagged trees	0.873	± 0.082	0.741	± 0.075	0.883	± 0.085	0.754	± 0.090	0.987	± 0.011	0.982	± 0.034
2	Int-Gabor	Bagged trees, med. G. SVM [1]	0.817	± 0.080	0.714	± 0.078	0.830	± 0.086	0.739	± 0.102	0.982	± 0.015	0.963	± 0.046
3	Int-GLCM	Bagged trees, fine G. SVM [1]	0.883	± 0.061	0.816	± 0.047	0.907	± 0.067	0.832	± 0.062	0.971	± 0.017	0.981	± 0.040
4	Int	Coarse KNN	0.624	± 0.071	0.523	± 0.059	0.721	± 0.091	0.585	± 0.082	0.826	± 0.033	0.841	± 0.051
5	Depth-Int-GLCM-Gabor	Boosted trees	0.929	± 0.070	0.847	± 0.064	0.941	± 0.076	0.863	± 0.072	0.988	± 0.014	0.983	± 0.056
6	Depth-Int-Gabor	Boosted trees	0.926	± 0.069	0.844	± 0.071	0.940	± 0.076	0.860	± 0.078	0.986	± 0.017	0.983	± 0.057
7	Depth-Int-GLCM	Fine G. SVM	**0.943**	**± 0.055**	**0.855**	**± 0.055**	**0.952**	**± 0.059**	**0.870**	**± 0.063**	**0.990**	**± 0.011**	**0.985**	**± 0.050**
8	Depth-Int	Fine G. SVM	0.935	± 0.059	0.853	± 0.067	0.951	± 0.066	0.869	± 0.073	0.984	± 0.021	0.984	± 0.057
9	DeepLabV3+ [2]	ResNet-101 [251]	**0.898**	**± 0.079**	0.729	± 0.081	**0.924**	**± 0.067**	0.810	± 0.070	0.968	± 0.027	0.877	± 0.056
10	DeepLabV3+ [2]	ResNet-50 [251]	0.894	± 0.087	0.733	± 0.078	0.916	± 0.081	0.814	± 0.067	0.973	± 0.023	0.880	± 0.061
11	Modified FCN [3,5]	ResNet-18 [251]	0.886	± 0.087	0.672	± 0.098	0.915	± 0.075	0.773	± 0.092	0.963	± 0.034	0.840	± 0.092
12	Modified U-Net [4,5]	N/A	0.895	± 0.087	**0.935**	**± 0.070**	0.912	± 0.084	**0.975**	**± 0.017**	**0.978**	**± 0.016**	**0.960**	**± 0.076**
13	U-MSAA-Net (proposed)	N/A	**0.977**	**± 0.040**	**0.945**	**± 0.067**	**0.983**	**± 0.041**	**0.981**	**± 0.018**	**0.993**	**± 0.009**	**0.964**	**± 0.072**

[1]: First classifier for 125 kHz, second classifier for 67 kHz.
[2,3,4]: Custom-trained DeepLabV3+ [244], Fully Convolutional Networks [243] and U-Net [217], respectively.
[5]: Employs *spatial consistency* checks: during upsampling, the spatial dimensions of feature maps are compared and slightly modified if necessary for matching.
†,‡: Binary semantic segmentation-based classifiers for the identification of krill and finfish in 125 kHz echograms, and exclusively finfish in 67 kHz echograms.

Table 4.2 compares the quantitative performance of the proposed and compared meth-

ods for the task of semantic segmentation on the test set of the FFK dataset (40 multi-frequency echograms). It is divided into three groups of configurations: #1 to #8 (compared traditional ML models), #9 to #12 (compared DL models), and #13 (proposed). The three reported metrics, IoU, precision, and recall, are computed on a pixel-level basis for each echogram and then averaged over the test set, for both the 125 and the 67 kHz frequencies. Bold fonts indicate the best performing models in each group of configurations.

For the compared traditional ML-based approaches (models #1 to #8), the best performing model is #7 at both frequencies, which includes the basic intensity and mean intensity information, depth information, and GLCM-derived features, using a fine Gaussian SVM (*i.e.*, with a kernel scale of 0.66) as a classifier. Models #5 to #8 significantly outperform models #1 to #4 in terms of both IoU and precision, indicating that depth information is crucial in helping remove a significant number of false positives detected by models #1 to #4 near the water-air interface (upper region of FFK samples). GLCM-derived features appear to be more suitable than Gabor filter banks for segmenting finfish and krill from echograms as they are involved in the best performing models, whether considering depth information (#7 and #5) or not (#3 and #1).

The compared DL-based approaches from models #9 to #11 use custom-trained versions of frameworks designed to segment cohesive groups of pixels (*e.g.*, classes from the PASCAL VOC 2012 dataset [252]). On the other hand, schools of finfish and swarms of krill often create particularly fine morphological structures. As a result, models #9 to #11 tend to over-classify pixels as belonging to the *positive* classes, as illustrated by their recall being consistently higher than their precision. The only DL-based method that obtains a precision higher than its recall (aside from the proposed) is the modified U-Net (#12) at 67 kHz, motivating its use as a main architectural reference of U-MSAA-Net. This tendency of higher recall is also observed for all ML approaches (models #1 to #8).

Conversely, the proposed U-MSAA-Net model (#13) is able to not only comfortably outperform, across all metrics, models #9 to #12, but also drastically close the gap between its own precision and recall. In fact, the precision of U-MSAA-Net for the more challenging 67 kHz echograms, 0.981, is higher than its recall of 0.964 for that same frequency. This result highlights the high level of reliability from the pixel-level output of the proposed system: it correctly identified, across frequencies, true positives in the test set in more than 98% of the cases. Although the performance

of the modified U-Net (model #12) is close to that of U-MSAA-Net, it yields considerable lower-quality predictions for 125 kHz, hindering its potential use with the multi-frequency FFK. Even though the 67 kHz recall of model #7 (0.985) is slightly higher than that of U-MSAA-Net (0.964), its precision is considerably lower: 0.870 compared to 0.981. This precision/recall gap from model #7 indicates that it is over-classifying pixels as *positive*; one should note that by simply considering every pixel as being from the positive class, any system can trivially obtain a recall of 1 in binary semantic segmentation tasks.

The similar performance observed for models #9 and #10—which used transfer learning and were trained from scratch, respectively—points to the fact that the non-natural appearance of the echograms in FFK mostly precludes the typical advantages of transfer learning. This indicates that the parameters learned by the DL models are particularly influenced by the non-natural-looking training data used and reach similar values after training, regardless of their initial states.

It is interesting to note the systematic difference in performance for all models but U-MSAA-Net, between 67 and 125 kHz. The sole presence of finfish in 67 kHz echograms creates structures that are way finer than those observed in aggregations of fish and krill (125 kHz). Consequently, the segmentation task for 67 kHz echograms represents a significantly harder challenge when compared to that posed by 125 kHz echograms (see GT masks in Figure 4.11). Thus, the aforementioned tendency of over-estimating pixels from the *positive* class results in better performance metrics for 125 kHz echograms across all models. The similar performance across frequencies of U-MSAA-Net (model #13) further demonstrates its ability to outperform state-of-the-art methods given different metrics and inputs.

4.6.3 Qualitative Comparison

Figure 4.11 shows sample representative results for three models from Table 4.2: the best compared traditional ML model (#7), a representative model from the compared DL group (#9) as there is no clear best model from that group, and the proposed U-MSAA-Net (#13). The two echogram examples are quite different in terms of the period of the year and the time of day (October late afternoon and July early morning). In the first example (top row), we can see, in addition to krill and schools of hake found below the krill layer in the lower third region, the following particularities: the presence of smaller aggregations towards the middle depth (vertically), which are

most likely juvenile hake, and a layer of zooplankton near the water surface (below the yellow line in the original 125 kHz echogram). In the second example (bottom row), krill and finfish appear intertwined in the bottom half region. It is worth mentioning that the ground-truth annotations are in no way "perfect": in the top example at 67 kHz, we can notice some horizontal patterns towards the bottom, which are most likely due to ringing from the 67 kHz transducer.

From the examples in Figure 4.11, we can see that the proposed approach (#13) produces results that are very close to the ground truth masks with a fine level of details, an observation that also generally applies to the best traditional ML model (#7). The compared DL model (#9), on the other hand, produces results that are coarser, missing many of the finer (smaller) aggregations. This is also true for the other DL architectures. Although models #7 and #13 appear to yield qualitatively similar results, upon closer inspection, the former tends to overpredict target species pixels. This can be seen in particular in the top row example at 125 kHz, where the large combined mixture of finfish and krill appears slightly (and incorrectly) denser in the predictions of model #7, compared to that of the proposed approach (#13). One may also note that both DL models (#9 and #13) successfully discard background pixels near or above the water-air interface that in some cases may look like marine species without the need for explicitly including depth information as input data; traditional ML approaches without explicit depth information (models #1 to #4) had difficulties with those pixels, detected as false positives, an issue resolved in models including depth information (#5 to #8). Overall, the proposed U-MSAA-Net yields the best qualitative results on the FFK dataset and in particular shows a substantial visual improvement over the compared DL model (#9).

4.6.4 Computational Considerations

Regarding the traditional ML approaches, the GLCM-derived features take the most time to extract, with an average total of 43.5 s/echogram for all four features, followed by the Gabor filter responses, with an average 4.2 s/echogram. The other features (intensity, mean intensity and depth) are either readily available or take negligible computational time (a few ms/echogram). The training times depend on the feature vector configuration and on the type of classifier. For the best model among traditional ML-based approaches (#7), which corresponds to a 7-dimensional feature vector paired with a fine Gaussian SVM for both frequencies, the training times were

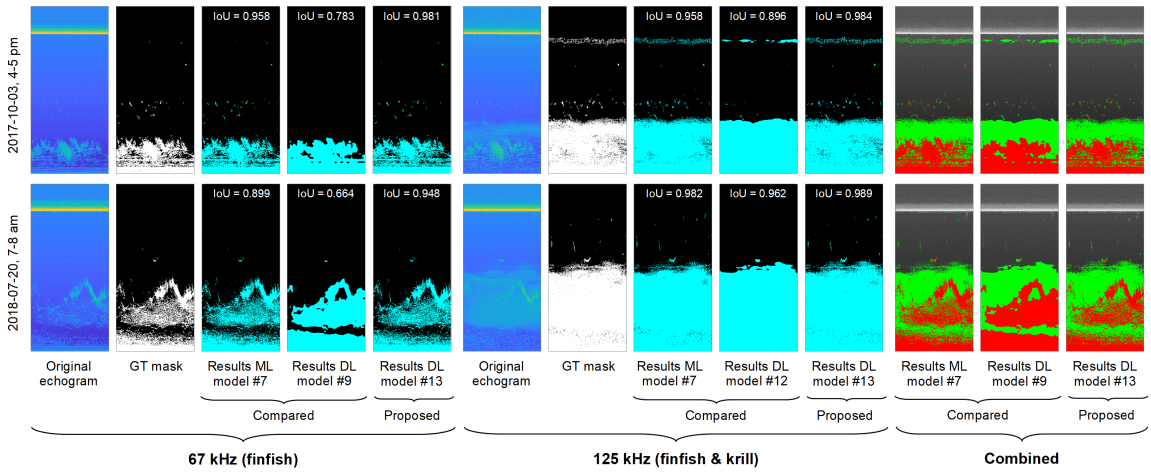


Figure 4.11: Sample semantic segmentation results for the best compared ML model (#7, *i.e.*, fine Gaussian SVM classifier with a feature vector including depth, intensity, mean intensity and GLCM information), a representative DL model from the compared DL group (#9, *i.e.*, DeepLabV3+ with a ResNet-101 backbone), and the proposed model (#13, *i.e.*, U-MSAA-Net). Combined results show detected finfish (red) and krill (green) overlaid on a grayscale version of the original 125 kHz echogram.

182 s and 147 s for the 67 and 125 kHz cases, respectively. The inference times (not including feature extraction) are 2.2 and 2.3 s/echogram for the 67 and 125 kHz cases, respectively, *i.e.*, a combined 4.5 s for a multi-frequency echogram.

The DL-based approaches are trained on a single GPU (see sub-section 4.6.1) with varying training periods. Their inference time, while almost negligible, is proportional to the number of trainable parameters of a model. Models #9 to #13 in Table 4.2 have, respectively, $61, 40, 11, 17$ and 18 million parameters (approximately). The fastest model (modified FCN) can process a pair of echograms (67 and 125 kHz) in 0.01 s while the slowest (DeepLabV3+ with ResNet-101) does it in 0.05 s. Our proposed approach (U-MSAA-Net) performs this multi-frequency segmentation in 0.02 s. This extremely low inference time allows for the timely DL-based processing of large amounts of acoustic backscatter data, in particular considering that each echogram might represent multiple minutes/hours (*e.g.*, in FFK, 1 hour). The ML models are significantly slower albeit still timely.

4.6.5 Applicability to Another Dataset: PLHS

Our research group previously introduced the PLHS dataset [76], as well as an instance segmentation approach to efficiently identify schools of salmon and herring on PLHS

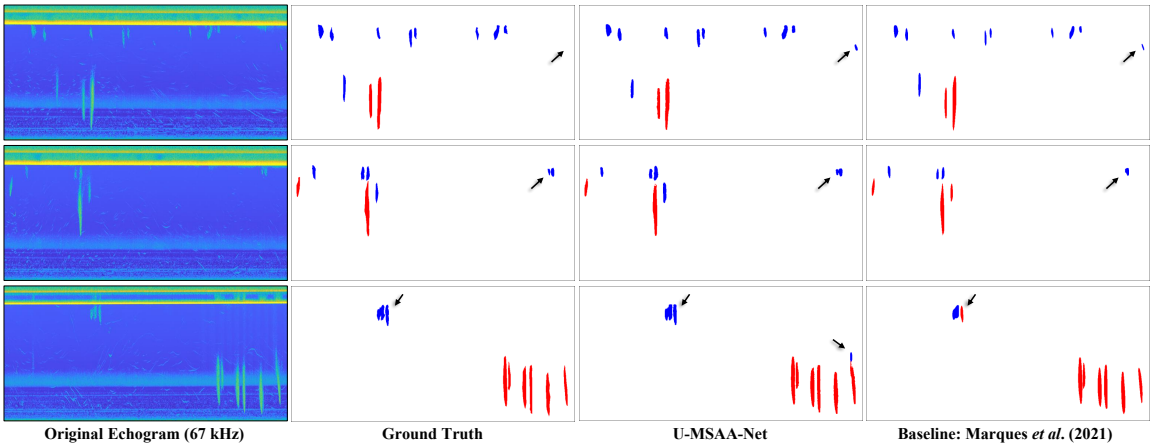


Figure 4.12: Sample semantic segmentation results on the test set of the PLHS dataset [76]. Schools of salmon and herring are represented in blue and red, respectively. Black arrows point to relevant regions of the images that are discussed in the text. Columns from left to right show, respectively, the PLHS echogram sample (67 kHz), pixel-level ground truth, output from U-MSAA-Net (model #16 of Table 4.3), and results from the instance segmentation baseline method our research group proposed in [76].

samples. In order to highlight the versatility of U-MSAA-Net with respect to various scenarios/datasets, we train it to perform semantic segmentation with the three-class (herring, salmon, background) PLHS. In contrast to the samples from FFK, the visual targets of PLHS (*i.e.*, schools of salmon and herring) present themselves as cohesive and more well-defined morphological structures, representing a suitable benchmark to study U-MSAA-Net's scalability. The detection performance reported in [76] is based on the pixel-level and bounding box-level intersection between *instances* of the ground truth and those predicted by the instance segmentation framework. Semantic segmentation does not distinguish between instances; thus, in order to compare the results of this baseline approach [76] with U-MSAA-Net, we calculate IoU, precision and recall metrics in an instance-agnostic and pixel-level manner for both methods (see Table 4.3).

Figure 4.12 shows three samples of PLHS (first column), their ground truth annotations (second column), U-MSAA-Net predictions (third column) and baseline predictions (rightmost column). A qualitative analysis of the predictions shows that even though U-MSAA-Net was not specifically designed for PLHS, it reaches and often surpasses the performance of the baseline method. In the first row of Figure 4.12, U-MSAA-Net and the baseline produced similar results: both methods correctly iden-

tified schools or herring (red) and salmon (blue), while indicating the presence of an extra school of salmon in the top-right portion of the echogram (as shown by the black arrows). The predictions on the PLHS sample in the middle row highlight two aspects: 1) U-MSAA-Net tends to create slightly larger predictions, which results in a considerably higher recall (see Table 4.3), and 2) the MSAA module allows the proposed method to identify smaller objects that are missed by the baseline (*e.g.*, one of the close-proximity schools of salmon in the top-right region of the sample as shown by the black arrows). The last row of Figure 4.12 illustrates a challenging scenario where numerous schools stand close to each other. U-MSAA-Net successfully identified (and distinguished between) all schools, while the baseline incorrectly misclassified a school of salmon (see top region, shown by the black arrows). On the other hand, the proposed method pointed to an invalid school of salmon in its output (*i.e.*, blue component on top of the rightmost school of herring, as shown by the black arrow).

Table 4.3: Semantic segmentation performance of U-MSAA-Net and baseline method on the multi-class single-frequency PLHS dataset [76].

#	Model	Epochs	Optm.	Learn. Rate	Dropout	Class	IoU		Precision		Recall	
14	Marques *et. al* [76]	5.6*	SGD	$2e^-2$	N/A	Salmon	0.477 0.246	±	**0.752** **0.329**	±	0.529 0.260	±
						Herring	0.758 0.090	±	**0.944** **0.065**	±	0.794 0.089	±
						Combined	0.618		**0.848**		0.661	
15	U-MSAA-Net	500	ADAM [250]	$1e^-3$	0.75	Salmon	**0.578** **0.245**	±	0.738 0.232	±	0.754 0.238	±
						Herring	0.772 0.095	±	0.846 0.101	±	0.908 0.093	±
						Combined	0.675		0.792		0.831	
16	U-MSAA-Net	500	ADAM [250]	$1e^-3$	0.50	Salmon	0.550 0.208	±	0.666 0.250	±	0.808 0.160	±
						Herring	0.811 0.065	±	0.893 0.063	±	0.904 0.079	±
						Combined	**0.680**		0.779		0.856	
17	U-MSAA-Net	1000	ADAM [250]	$1e^-3$	0.75	Salmon	0.532 0.219	±	0.712 0.239	±	0.722 0.210	±
						Herring	**0.813** **0.071**	±	0.897 0.060	±	0.904 0.089	±
						Combined	0.673		0.804		0.813	
18	U-MSAA-Net	500	SGD	$1e^-3$	0.5	Salmon	0.518 0.188	±	0.585 0.221	±	**0.854** **0.141**	±
						Herring	0.787 0.076	±	0.841 0.089	±	**0.932** **0.071**	±
						Combined	0.652		0.713		**0.893**	

*: The model our research group proposed in [76] is trained for 300 2-sample batches, which equates to 5.6 epochs of the PLHS dataset.

Table 4.3 compiles the IoU, precision and recall of U-MSAA-Net and the baseline method on the test set of PLHS. The combination of relevant U-MSAA-Net hyperparameters that led to a particular performance is also presented for reference. As shown in Figure 4.12, the baseline method (model #14) typically predicts smaller pixel groups, leading to the highest precision in parallel with the smallest recall.

The best recall is obtained by model #18, where U-MSAA-Net was trained for 500 epochs with a stochastic gradient descent optimizer (SGD, which takes longer than ADAM [250] to achieve local and global minima), resulting in a less-specialized detector that overpredicts pixels as foreground (*i.e.*, high recall and lower precision). Models #15, #17 and #16 offer a mix of good precision and recall, while respectively yielding the best IoU performance for the classes salmon, herring and a combination of them. These three models are trained with ADAM, pointing to the fact this optimizer results in detectors with an excellent balance of completeness (recall) and certainty (precision and IoU). Overall, Table 4.3 indicates that the specific needs of a user can be met by the same model under different training schemes: model #18 is recommended for analyses where the identification of all specimen is important and false positives are tolerable; models #15 and #17 are suited for more salmon- and herring-focused studies (respectively), whereas model #16 offers a good trade-off between the two classes.

4.7 Conclusion

This study addresses the pixel-level detection of co-occurring finfish and krill from multi-frequency echograms via a novel DL semantic segmentation network: U-MSAA-Net. U-MSAA-Net, a U-Net-based framework that incorporates Multi-Scale Additive Attention (MSAA) modules, allows for an efficient suppression of the feature responses from image regions with lesser semantic value, due to its leveraging of contextual and local information from feature maps available at any given level of the decoding phase of the network. The proposed strategy of pixel-level classifications at multiple frequencies allows for the extraction of the diffuse layers of krill mixed with schools of finfish (particularly hake) at 125 kHz and for the extraction of schools of finfish at 67 kHz, with krill pixels deduced as the difference in the results. The study also provides a comprehensive experimental comparison that covers both ends of the learning spectrum, including traditional ML methods based on texture features and classical classifiers, and DL methods based on various state-of-the-art semantic segmentation networks. Echogram annotations at the pixel level, performed in a semi-automatic fashion, are made simpler and more efficient via a modified adaptive Wiener filtering stage that produces SNR images.

Experiments on the FFK dataset of 200 echograms covering a large variety of samples yield promising results, with U-MSAA-Net overall outperforming all com-

pared methods: IoU values up to 0.977, precision values up to 0.983 and recall values up to 0.993. Compared traditional ML methods tend to generate results at a finer level of detail albeit at a larger computational cost time-wise, whereas compared DL methods tend to focus on the general trends by offering results at a coarser level of detail, with shorter inference times. U-MSAA-Net offers the best of both worlds, with fine-grained results and a low time-wise computational cost. Its inference time at the centisecond level opens up the way not only for real-time analyses (as each echogram represents several minutes/hours, one hour in the case of FFK) but also for a timely processing of legacy data. Additional experiments on the PLHS dataset of schools of herring and salmon showcase U-MSAA-Net's versatility in terms of scenarios and datasets, with U-MSAA-Net able to provide a similar or superior performance compared to the instance segmentation network that was originally developed for PLHS. In particular, the MSAA module allows for the identification of a wider range of objects size-wise. The architecture of U-MSAA-Net also allows for the easy processing of input echograms of any size, a sizable advantage when dealing with acoustic data for which the relevant time window and depth of the water column varies depending on the species of interest.

Future work could look at expanding the detection to other species and phenomena for a more comprehensive tool that could support stock and ecosystem assessments.

4.8 Acknowledgments

The work described in this Chapter was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and ASL Environmental Sciences through the Alliance Grants program. Generous in-kind support in the form of active acoustic Environmental Monitoring data and biological expertise was provided by Canada's Department of Fisheries and Oceans.

Chapter 5

Size-invariant Detection of Marine Vessels from Visual Time Series

This chapter details a novel system proposed for the identification of marine vessels with heterogeneous visual characteristics (*i.e.*, size, color, shape, level of visibility and occlusion) in monitoring images. The hybrid approach employed takes advantage of state-of-the-art object detectors for the identification of medium- and large-sized vessels, while also employing "traditional" Computer Vision methods for detecting hard-to-see small vessels. The contributions of this work consist of two datasets of monitoring images containing marine vessels, a novel *bidirectional* Gaussian Mixture Model (GMM)-based motion detector, a vessel detector bounded by four *environmental* assumptions, and a custom-trained image classifier specialized in small visual targets.

The approach proposed in this chapter, as well as its detailed evaluation, also allows for a reflection on the use of CV systems for the processing of out-of-water Environmental Monitoring visual data (see Table 1.2). The efficiency of this novel detector is analyzed via numerous experiments (Table 5.2) involving thousands of Environmental Monitoring images acquired by University of Victoria's Coastal and Ocean Resource Analysis Laboratory (CORAL) off the coast of Vancouver Island.

The content of this chapter was submitted, peer-reviewed in a double-blind manner, approved, and presented at the 2021 Winter Conference on Applications of Computer Vision (WACV) [56]. Minor editorial changes were done to the content of [56]. Two figures that were originally omitted from the manuscript in [56] because of space constraints, 5.3 and 5.9, are also included in this chapter.

5.1 Introduction

Anthropogenic activities in coastal areas (*e.g.*, vessel traffic, fishing, recreation) can inflict long-lasting harmful effects on oceans. Given that sound travels five times faster in water than in air [253], noise pollution is correlated to behavioural disturbance in marine species [254, 255] and interferes with animal vocalization [256, 257]. The ecological footprint from increased marine vessel traffic observed over the last few decades [258] is clearly demonstrated [259].

Automatic Identification System (AIS) data from marine vessels is used for estimating vessel densities as spatially explicit proxies for stressors such as noise, disturbance, vessel-strike, and discharge of harmful substances [260, 261]. However, AIS was not designed as a tool for research and conservation [262], thus impact estimates based solely on it often ignore contributions from smaller vessels (which typically do not broadcast AIS data [263]). This poses a challenge for understanding the contribution of smaller vessels to the marine soundscape, particularly in densely populated coastal regions like the Salish Sea.

Among the several cetacean species found in the inshore waters of the Salish Sea are the endangered [264, 265] Southern Resident Killer Whales (SRKW); the majority of the Salish Sea is currently designated as a SRKW critical habitat. This area is highly trafficked by small whale-watching, fishing, research and recreational boats. Research shows that these vessels emit high acoustic energy in the mid- and high-frequency ranges, and are more likely to negatively interact with sensitive marine life [266]. Therefore small boat traffic has wide-ranging ecological impacts [266, 267, 268].

Optical systems [269, 270, 271] complement well AIS-based monitoring because of their ability to detect both AIS and non-AIS boats, their non-invasive nature and low-cost [272]. Visual sightings might also provide additional information such as the type of interaction a vessel engages in with the environment. However, the interpretation of these visual data is time-consuming [273, 199]. Large amounts of monitoring data [272] require manual detection in tedious and often error-prone routines, the reason why numerous recent works offered automatic vessel detection frameworks [274, 275, 276, 277, 278, 279, 280].

While recent DL-based object detectors [281, 282, 118] are efficient in identifying large-sized or near-shore marine vessels, they often miss small boats (either farther away from the camera or because of their actual size) [263]. For this reason, our study

aims for the automatic detection of marine vessels of *any* size in visual data obtained at two sites inside the SRKW's critical habitat (see 5.4.1).

The remainder of this Chapter is structured as follows. Section 5.2 discusses works related to the proposed system. Section 5.3 details our approach for the detection of boats of heterogeneous visual characteristics. In Section 5.4 we introduce two annotated datasets of images from monitoring sites in the Salish Sea and use them to evaluate the proposed system with respect to state-of-the-art object detectors [281, 282, 118]. Section 5.5 draws concluding remarks.

5.2 Related Works

Relevant works to our approach include custom detectors of marine vessels and generic DL-based object detectors.

Marine Vessel Detection. Methods that perform the visual detection of boats have been proposed for a number of monitoring configurations, such as satellites, Unmanned Aerial Vehicles, boat-attached cameras and fixed-position cameras. Elvidge *et al.* [274] used infrared satellite images to detect boats in the day/night band based on the assumption that the lighting sources generated by fishing boats can be identified as intensity spikes. Using a different optical system layout, Kruger and Orlov [275] mounted a thermal imaging system on autonomous platforms and used its data to detect small vessels. Their method first estimates horizon lines, then performs detection in the vicinity of these lines, followed by tracking the identified objects.

Tran and Le [276] performed boat detection in sequences of images from a fixed location. Their first step, *temporal attention*, executes a background subtraction that isolates only moving elements of the sequence. A parallel step, *spatial attention*, looks for the salient regions of the image sequence. The final output is a weighted linear combination of both steps. The authors note that the segmentation of aquatic background (so that the foreground highlights only marine vessels) on *in-situ* surveillance images or videos is a challenging task, given the waters dynamic nature. Bao *et al.* [277] propose to first use a graph-based segmentation to detect water, followed by a saliency-based vessel detection. Bloisi *et al.* [283] offered a method that discretizes an unknown distribution, ultimately aiming to describe highly dynamic backgrounds (such as the surface of the water).

So far DL has been only sparsely used for marine vessels detection. Tang *et*

al. [278] used a custom Artificial Neural Networks (ANN) to extract and classify candidate ship features. Liu *et al.* [279] proposed rotated region convolutional neural networks (RR-CNN) to identify marine vessels in satellite images. RR-CNN are able to encompass rotated targets under Region of Interest (ROI), but none of the vessels in the dataset used by the authors (HRSC2016[1]) are small (as opposed to our **D2** dataset; see 5.4.1). Zhang *et al.* [280] extracted handcrafted vessel features from satellite images using line segments and saliency maps, and employed Convolutional Neural Networks (CNN) to classify them.

Generic Object Detection. Object detectors perform both localization and classification tasks. Until 2012, most object detection methods (described as "traditional detection methods" in [284]) extracted and used handcrafted features such as the Histogram of Oriented Gradients [285], multiple feature extractors [110, 112, 115], or class- and application-specific features [286, 287]. The Deformable Part-based Model (DPM) developed by Felzenszwalb *et al.* [288] and its further developments [289, 290] are considered the best-performing methods among the traditional detection methods.

AlexNet, a CNN-based image classifier proposed by Krizhevsky *et al.* [291], demonstrated the potential of using CNNs to extract generic and highly discriminative features from large sets of data [246]. Subsequently, a number of works proposed CNN-based object detectors capable of delimiting *where* target objects are, and exactly *what* class they belong to: R-CNN [292], Fast R-CNN [293], Faster R-CNN [118], Cascade R-CNN [282] and RetinaNet [281], among others [294, 295, 296, 297]. This process typically involves the training of CNN-based modules that perform localization and classification individually ("two-stage detectors") or under the same trainable network ("one-stage detectors"). Lin *et al.* proposed Feature Pyramid Networks (FPN) [297], capable of integrating the representation under multiple scales of objects during the CNN training process. When used with end-to-end object detectors, FPN significantly increases the final detection performance.

To the best of our knowledge, no work has used state-of-the-art object detectors pre-trained on large datasets [247] as part of a marine vessel detection framework. Moreover, we propose a novel system that also focuses on the identification of small vessels observed in land-based visual data, otherwise ignored by object (and more specifically marine vessels) detectors.

[1] www.kaggle.com/guofeng/hrsc2016

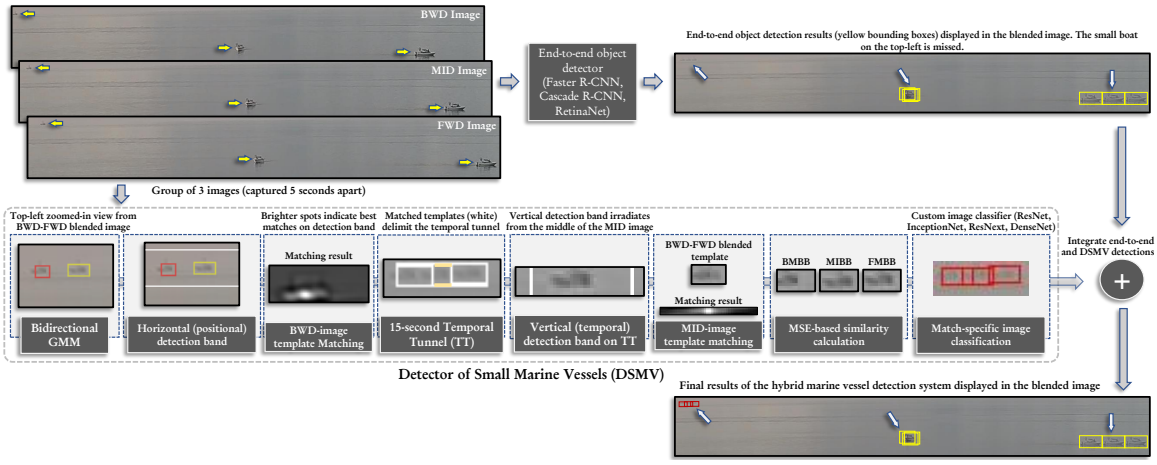


Figure 5.1: Hybrid marine vessel detector proposed. The detection results of the end-to-end object detector and the DSMV are combined for enhanced detection capabilities. We invite the reader to zoom-in on this and the other images of this Chapter.

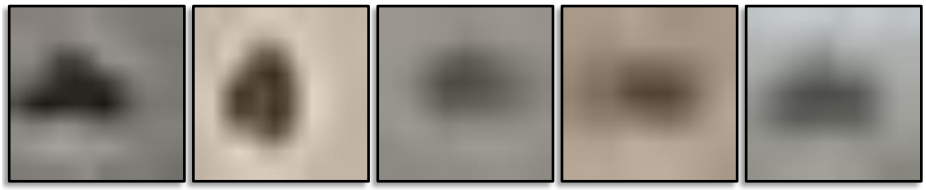


Figure 5.2: Examples of small marine vessels (mean area of 79 pixels) resized to 224×224 pixels. See sub-section 5.4.1 for details.

5.3 Proposed Approach

We propose a hybrid marine vessel detection system that combines state-of-the-art object detectors and a novel Detector of Small Marine Vessels (DSMV). The DSMV uses short time series of images (*i.e.*, three images at a time) to detect small vessels. We use the term *small* to refer to vessels that have approximately 80 pixels of area, while *medium* and *large* vessels are those that occupy approximately 800 and $1,500$ pixels of area, respectively (see 5.4.1). The proposed system does not require a (often most error-prone) sea-land segmentation step as so other works [278, 277, 283]. It thus contributes not only to the Environmental Monitoring field, but also addresses the more general challenge of *small object detection*. Figure 5.1 visually summarizes the hybrid detection approach proposed.

Figure 5.2 highlights a fundamental challenge for the identification of small boats: given their size and changing appearance, a regular CNN-based feature extractor and classifier might not be able to distinguish them from a highly dynamic background that includes water surface perturbations, sunlight reflection or weather elements, floating driftwood and kelp. Indeed, the visual structures that compose these background elements are nearly identical to that of boats in many instances. We thus rely upon temporal information (*i.e.*, movement conveyed by multiple images/frames of the same scene) as a cue for the presence of boats. We assume that a boat is going to move in a roughly horizontal manner (considering fixed-position cameras) and that its visual features are not going to change during a small, 15-second time window.

DSMV considers only three 5-second-apart images of the same location (henceforth referred to as "BWD", "MID", and "FWD" images) obtained from a land-mounted camera. This temporal window is chosen so that the DSMV can be deployed in remote sites where only limited data storage and transmission capabilities are available. Based on a thorough analysis of visual EM data of coasts, we define four *environmental* assumptions that help distinguish small marine vessels from false positives: **A1**) vessels that appear small on monitoring images are those farther away from the camera, and thus they should only move horizontally in a 15-second time window; **A2**) a boat identified in the MID image will remain inside a sub-region of this image bounded by the same boat identified in the BWD and FWD images; **A3**) the boat in the MID image is going to be positioned roughly in the middle of the aforementioned sub-region; **A4**) the three sightings of the same boat identified in the three images exhibit similar visual appearance.

These four assumptions encode paramount contextual information acquired from the study of monitoring data. We explicitly incorporate them into the DSMV with the use of traditional Computer Vision (CV) methods, as detailed in the remainder of this Section. In order to further refine the detection results, we include a custom DL-based image classifier at the end of the DSMV. As a result, DSMV performs robust detection combining specific, contextual data, and generic visual features learned via the training of CNN-based frameworks. Notes about the implementation of the proposed hybrid detection system are concentrated in sub-section 5.4.3.

Bidirectional GMM. Traditional GMM-based systems [298] typically create background models based on inputs I ranging from $I_1, ..., I_{t-1}$, and perform foreground detection on the current input, I_t, in an overall strategy that we will henceforth refer to as *forward motion*. In the proposed system we derive an exclusive GMM for each

group of three images (*i.e.*, BWD, MID and FWD), allowing for systems with low monitoring frequency to still use the proposed DSMV (as each group of inputs is processed independently). During the *forward motion*, the BWD and MID images are used to model the background, thus detecting the foreground on the FWD input. Similarly, in our novel *backward motion* strategy we use the FWD and MID images to detect motion on the BWD input (see Figure 5.3).

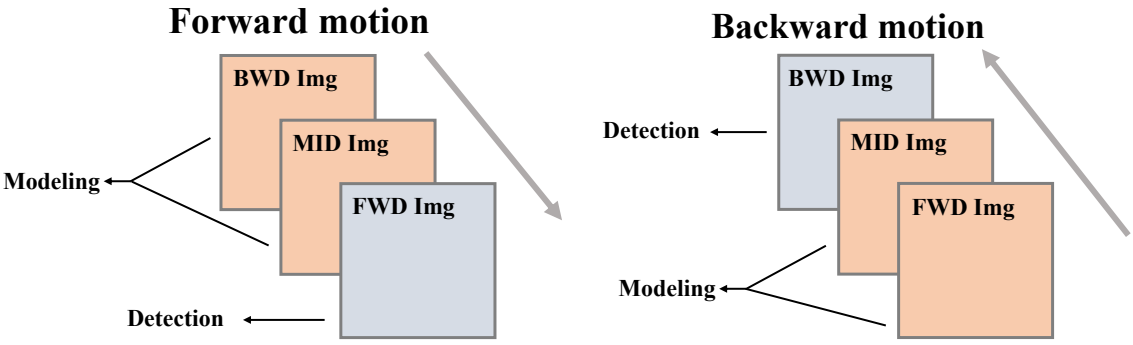
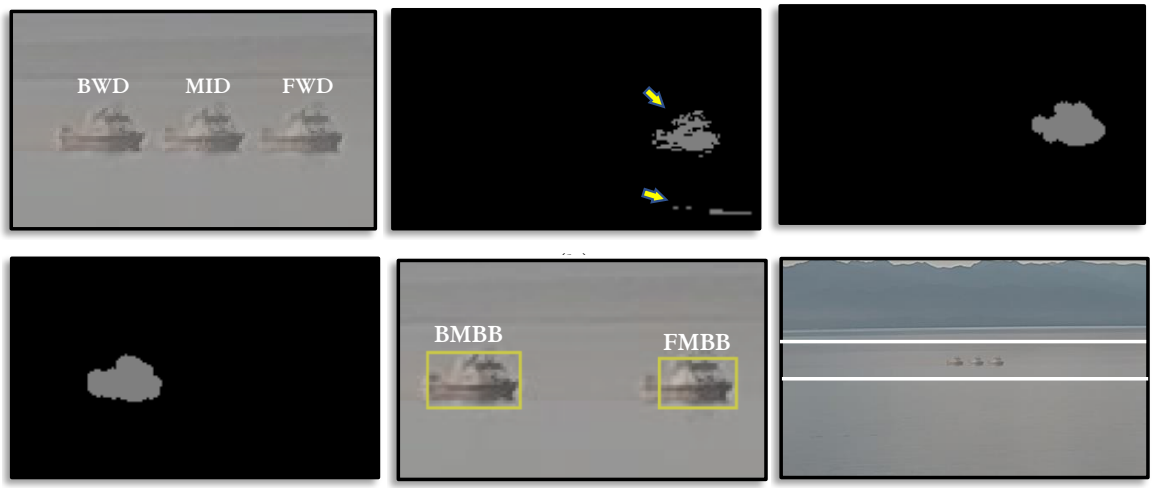


Figure 5.3: **Left**: *forward motion*; regular GMM strategy considering inputs for modeling and detection in chronological order. **Right**: *backward motion*; in the proposed approach, we also consider the two last inputs as the basis of a second GMM to detect motion-triggered elements in the first input (*i.e.*, BWD image).

A similar approach was proposed by Shimada *et al.* [299], where two models are derived from distinct groups of "past" and "future" frames that do not overlap. Also, Minematsu *et al.* [300] uses two models derived from the same group of past frames which are analyzed in regular and backward chronological order. Our approach is different because we consider all temporal information by modeling two GMM out of an overlapping frame (*i.e.*, MID image), and use both forward and backward motions.

Figure 5.4 illustrates our bidirectional GMM approach. In the forward motion, a set of motion-triggered connected components indicate the pixels that deviated from the background models created with the BWD and MID images (Figure 5.4b). Since the vessels are expected to create larger groups of quasi-connected components, we filter the results using morphological operations: an opening with a 3×3 ellipse followed by a dilation with a 5×5 ellipse. As shown in Figure 5.4c, this filtering eliminates small motion-triggered outputs (mostly noise) and combine the remaining pixels into compact groups. Figure 5.4d illustrates the result from the same process for the backward motion. The sets of connected components from each motion are used to delimit bounding boxes (BB) in the FWD and BWD images (Figure 5.4e), named forward-motion bounding boxes (FMBB) and backward-motion bounding boxes (BMBB).



(d) (e) (f)

Figure 5.4: Bidirectional GMM strategy proposed. **(a)** Blended section from a group of three images. **(b)** Forward motion raw output. **(c)** Forward motion filtered output. **(d)** Backward motion filtered output (BWD image). **(e)** Bounding boxes encompassing motion-triggered results. **(f)** Horizontal (positional) detection band.

Following assumption **A1**, we set a horizontal detection band (Figure 5.4f) inside which the marine vessel is expected to travel through the group of three images. The height of this band is a product of a configurable parameter ψ_{hdb} by the height from each FMBB (see Figure 5.5). Each FMBB determines a horizontal detection band where template matching is performed.

Template matching. A FMBB verifies **A1** if a BMBB exists on the BWD image inside the horizontal (positional) detection band set by its position, height and a given ψ_{hdb} (see Figure 5.5). Template matching operations are only performed on pairs of valid FMBB/BMBB positioned inside the horizontal detection band, as illustrated by BMBB match candidates 1 and 2 of Figure 5.5. There are a number of approaches to follow when a query image has to be found/matched in another image. Most commonly, one would start by determining visual features using a feature extractor (*e.g.*, SIFT [110], SURF [112], ORB [115]) and then match features between queries and candidates. However, since 1) we only compare each FMBB with a few potential BMBBs placed inside a reduced detection band, and 2) small regions representing boats often do not generate any output from regular visual feature extractors; we use template matching as in Kaehler and Bradski [301], which is simple and fast. The dimensions of a matched BMBB are adjusted to be equal to its template FMBB.

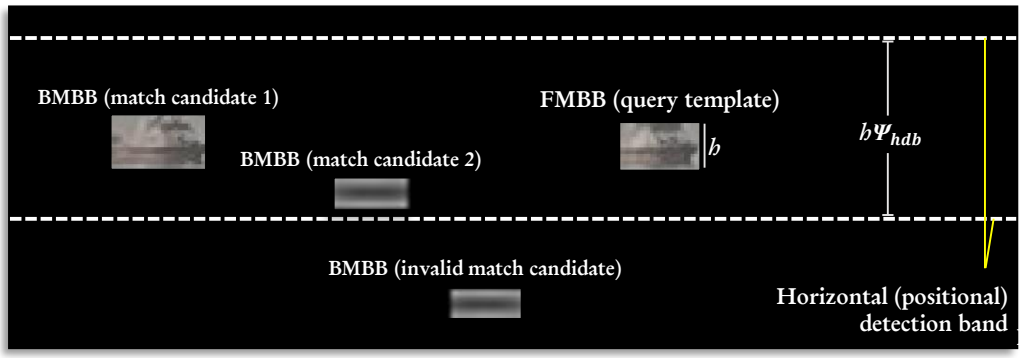


Figure 5.5: FWD-BWD images template matching. Each FMBB determines a single horizontal detection band based on their position, height h and parameter ψ_{hdb}. If one or more BMBB are positioned inside this band, the one that better matches the current FMBB is considered as its match.

Temporal Tunnel (TT). Once a FMBB is matched with a BMBB, assumption **A2** states that another match of that same marine vessel exists in the MID image, placed inside a TT delimited by the FMBB/BMBB pair (see Figure 5.6a). More specifically (**A3**), the vessel in the MID image should sit roughly in the middle of the TT. The width of this valid matching area in the middle of the TT is a product of width w from the FMBB by a configurable parameter ρ_{vdb} (see Figure 5.6c). A blended template (Figure 5.6b) is created by combining the BMBB and FMBB to prevent eventual occlusions in either reference BB from interfering with the MID-image matching. The blended template is used in a matching process that covers the entire TT (not only the valid matching area), resulting in a best match for MID-image bounding box (MIBB). If the MIBB falls inside the valid match region (green portion of Figure 5.6c), it is considered to be a valid candidate, as illustrated in Figure 5.6d.

Similarity Criteria. Assumption **A4** is based upon the empirical observation that within a 15-second time window all sightings of the same vessel should present similar visual appearance. This helps to further distinguish valid detection from false positives triggered by weather or incorrect matching results (see Figure 5.7 right). Each group of BMBB-MIBB-FMBB is used as the input of a Mean Squared Error (MSE)-based similarity analysis. The MSE between the blended FMBB/BMBB template (*e.g.*, Figure 5.6b) and the MIBB is measured, and if it does not exceed a threshold MSE_{th}, this group of three bounding boxes is further analyzed by an image classifier (see Figure 5.7).

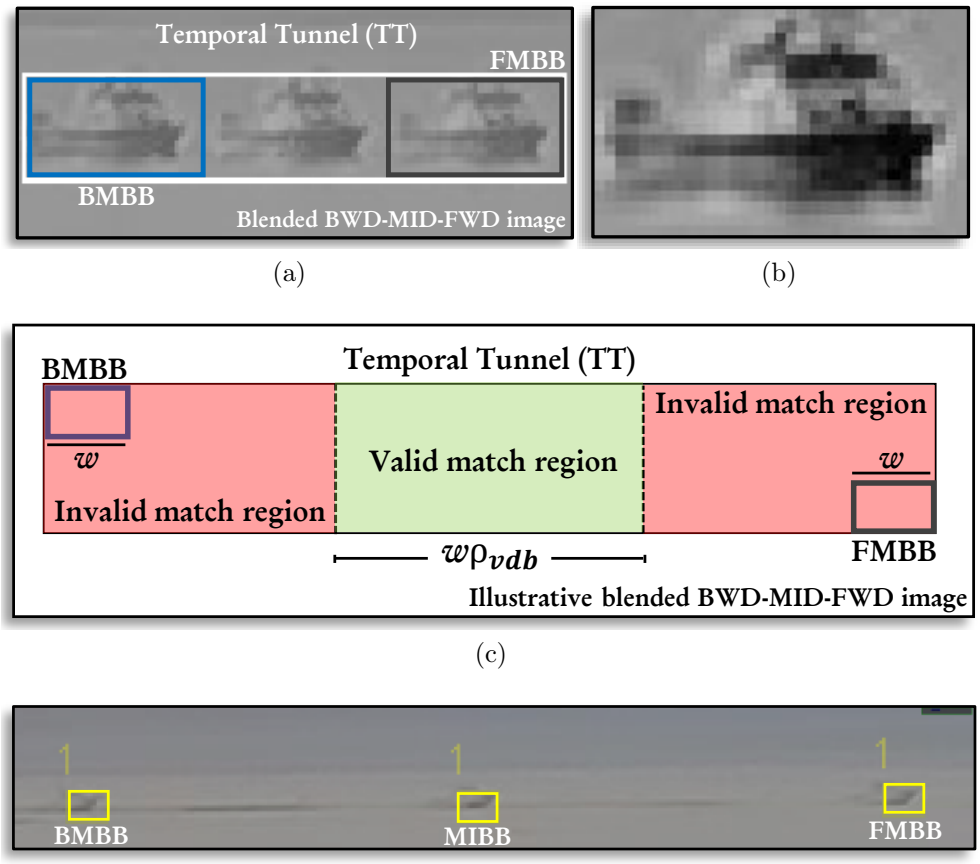


Figure 5.6: Temporal Tunnel (TT). **(a)** A 15-second TT bounded by a FMBB/BMBB pair. **(b)** Blended template composed by the FMBB and BMBB contents. **(c)** Only the template matching results inside a sub-region of the TT delimited by parameter ρ_{vdb} and width w are valid (**A3**). **(d)** BMBB-MIBB-FMBB matching output.

Image classification. The last step of the DSMV uses a custom-trained DL-based classifier (we evaluate six state-of-the-art options, see 5.4.2) to classify each individual bounding box in a group of BMBB-MIBB-FMBB. A group of BMBB-MIBB-FMBB is only deemed as valid if the content of all three bounding boxes is classified as an object of the *vessel* class. The custom-trained DL-based system performs a binary classification where each image patch is determined to belong to either the *background* or *vessel* class (as illustrated by Figure 5.8b). We train the DL classifiers by running the DSMV (without this final image classification step) on $1,644$ monitoring images (Figure 5.8a), and manually distinguishing between vessels and background patches in the resulting BMBB-MIBB-FMBB groups. These manually-

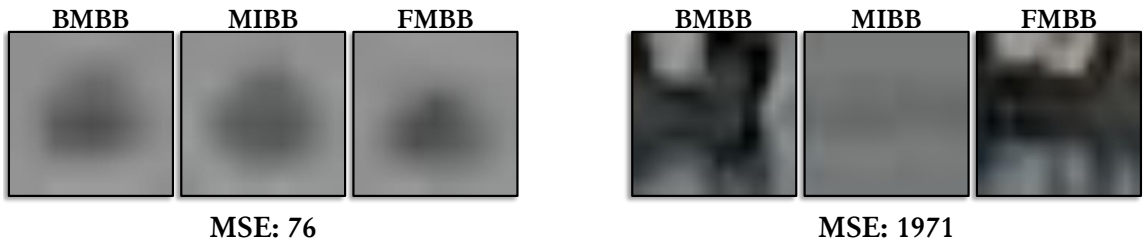


Figure 5.7: MSE-based similarity calculation. **Left:** Valid group of detection resulting in a low MSE. **Right:** An invalid group of detection candidates identified by a higher MSE. The similarity is measured between the contents of the blended FMBB/BMBB (see Figure 5.6b) template and the MIBB.

curated image patches are resized to comply with each classifier's CNN layout (*i.e.*, either 224×224 pixels or 299×299 pixels for the architectures considered) and used in the training routine, as shown in Figure 5.8b. This training process uses images obtained off the coast of Vancouver Island, Canada, during the years of 2019 and 2020 (see Table 5.1). In total we used $1,879$ vessel image patches ($1,544$ train/335 validation) and $2,264$ background image patches ($1,633$ train/629 validation). Figure 5.8c illustrates a scenario where false positives are accurately classified as background (yellow bounding boxes).

End-to-end object detection. State-of-the-art object detectors are typically trained on large datasets (*e.g.*, COCO [247], ImageNet [246]) that include one or more classes related to marine vessels. Thus we use pre-trained end-to-end object detectors (see Section 5.4) to find easier-to-identify boats, represented by medium- and large-sized vessels. Given the efficacy of such systems, we combine the output of object detectors (medium- and large-sized boats) and the DSMV (small-sized boats) into a robust hybrid detector capable of identifying boats of diverse sizes (see Figure 5.1).

Smaller vessels represent a challenging task to end-to-end object detectors because their feature extractors are based on multiple layers of sequential convolutions that generate feature maps of progressively smaller spatial dimensions. Small visual targets in the original image disappear during the feature extraction process (*i.e.*, they are represented by less than a pixel in the feature map at a certain depth in the CNN), preventing their localization and classification. Figure 5.9 shows that end-to-end object detectors can efficiently identify larger vessels (white arrows), but often miss smaller ones (yellow arrow).

Figure 5.8: Final step of the DSMV: custom-trained image classification. **(a)** Image from the training set with vessels highlighted. **(b)** Resized patches used in the training of image classifiers. **(c)** DSMV results with a custom-trained ResNet-50 [251] distinguishing between groups of valid vessels (red BBs) and background (yellow BBs).

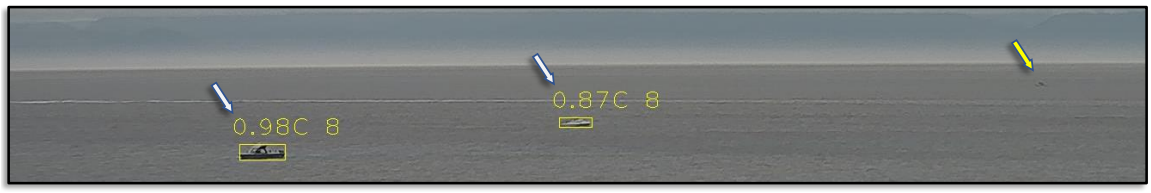


Figure 5.9: Detection results from a pre-trained end-to-end object detector (Faster R-CNN [118]). While medium-sized boats are detected with high confidence scores (0.98, 0.87), the yellow arrow shows that a small boat was not recognized.

5.4 Experimental Results and Discussion

This section details the experimental settings and results from a performance evaluation where two novel datasets (**D1** and **D2**) are introduced and used to compare our proposed method with five state-of-the-art end-to-end object detectors.

5.4.1 Marine Vessels Datasets

The images used in this project were obtained by University of Victoria's Coastal and Ocean Resource Analysis Laboratory (CORAL)[2] using optical cameras focused offshore to the south and west of southern Vancouver Island, BC, Canada, during the years of 2019 and 2020. These monitored regions of the Salish Sea are classified as SRKW critical habitats. A LOREX® pan-tilt-zoom (PTZ) camera was installed at two fixed positions on a headland overlooking a major vessel traffic thoroughfare and configured to continuously capture three 1920×1080 pixels photos in the first 15 seconds of each minute. This time-lapsed configuration allows for the inference of vessel movement, directionality and behaviour.

We manually annotate (*i.e.*, bounding boxes are drawn around marine vessels) and make publicly available[3] two datasets used to evaluate the proposed hybrid detector under two conditions: **D1**) 633 images containing boats of various sizes (mean vessel area of 953 pixels); **D2**) 138 images presenting only small boats with a mean area of 79 pixels. **D2** highlights the capabilities of the DSMV, as most of its marine vessels are missed by the state-of-the-art object detectors. While creating both **D1** and **D2** we selected images under different weather conditions (see Figure 5.10a) and vessel layouts, so that all monitoring scenarios are well represented. Note that a dataset of vessel patches, **D3**, is also created exclusively for training the last step of the DSMV (see Figure 5.8b), image classification. Samples from the training and testing datasets are never obtained from the same day (see Table 5.1), thus avoiding any data contamination during the evaluation process.

Dataset	Purpose	Description	Images	Vessels count	Dimensions	Dates
D1	Testing	BBs around vessels of various sizes	633	1056	1920×1080	2019: Sep 1,2,4-5,20 Aug 18
D2	Testing	BBs around small vessels	138	165	1920×1080	2019: Jul 30-31
D3	Training / Validation	Square vessel & background patches for training	3,177/ 964	1,879	CNN-dependent†	2019: Jul 27-29 Sep 6-11,16-19; 2020: Jan 1

†: Patches of 299×299 pixels for Inception [302], 224×224 pixels for DenseNet [200], ResNet [251], ResNext [303] and Wide ResNet [304] (see Figure 5.8b and Figure 5.2).

Table 5.1: Datasets for training the DSMV image classifier and evaluating the overall hybrid detection system proposed.

[2]www.coral.geog.uvic.ca

[3]https://github.com/tunai/hybrid-boat-detection

(a) Hazy-day image from **D1** showing two medium-sized vessels.

(b) Clear-day image from **D2** containing only a small vessel.

Figure 5.10: Datasets used for testing purposes. **D1** is composed by medium- and large-sized vessels (a), while **D2** (b) offers exclusively small-sized boats.

5.4.2 Experimental results

We carry out experiments comparing the performance of the proposed hybrid system with that of five pre-trained state-of-the-art object detectors. We also use six custom-trained image classifiers to study the capabilities of the DSMV given different CNN architectures.

The performance evaluation uses **D1** and **D2** and starts by employing five state-of-the-art end-to-end object detectors: Cascade R-CNN [282], Faster R-CNN [118] (with three different feature extraction networks), and RetinaNet [281]. All detectors used employ FPN [297] in their feature extraction routines. These detectors are pre-trained on the COCO [247] dataset, and since one of its 80 classes represents marine vessels ("boats" class), the initial set of experiments takes advantage of the pre-trained weights of these object detectors (see Table 5.2, configurations #1 to #5), by looking only at detection of this class. Transfer learning experiments where we re-trained only part of these detectors using our custom detasets could not surpass the performance of the pre-trained weights. Thus the following results for end-to-end object detectors

reflect the use of such weights.

The second part of the experiments (*hybrid* layout) evaluates the performance of the hybrid detector proposed. Given that numerous smaller vessels are missed by the end-to-end object detectors, combining their output with those from the DSMV greatly enhances the detection performance, especially for **D2** (where the vessels are particularly small).

We report the average precision (AP) in range $[0, 1]$ for three different Intersection-over-Union (IoU) thresholds $\in [0.2 : 0.1 : 0.4]$ (see Table 5.2). The decision of using lower-than-usual thresholds is based on the fact that the monitoring systems expected to use the proposed hybrid detector do not prioritize a precise fit around the visual targets, but rather a robust identification of their presence.

Each hybrid layout explores the performance of the DSMV using one of six custom-trained image classifiers: ResNet-50 [251], Inception-V3 [302], DenseNet-201 [200], ResNext-50 and ResNext-101 [303], and Wide ResNet 50-2 [304]. The detection time per image using a PC equipped with an Intel® Core i7-9700 CPU, 32 GB of RAM memory and a GeForce® GTX 1660 Ti GPU is approximately 0.2 seconds when using only end-to-end object detectors, and roughly 0.4 seconds for the entire hybrid approach.

Table 5.2 presents the detection results for **D1** and **D2** for both end-to-end object detectors and the proposed hybrid approach. Due to space constraints we present only the best-performing results out of the 35 configurations tested. The first five configurations use only object detectors, and among these RetinaNet performed significantly better for vessels of various sizes (**D1**). The best-performing object detector for IoU thresholds 0.3 and 0.4 using **D2** was Faster R-CNN, showing that the dataset composition and IoU threshold must be considered when choosing which pre-trained object detector to use. Since **D2** presents boats on average 12 times smaller than those in **D1**, the detection task becomes much more challenging, as reflected by the lower performance of the pre-trained object detectors in the **D2** dataset.

The proposed hybrid approach (*i.e.*, configurations #6 to #26 on Table 5.2) improved the performance from all state-of-the-art object detectors when corresponding stand-alone and hybrid layouts were compared. For example, the performance on dataset **D1** of Cascade R-CNN using ResNet-50 and IoU threshold 0.2 (configuration #4) was boosted in 16.45% by the addition of the DSMV employing ResNext-101 as backbone (configuration #14). Although a better performance is always provided by the proposed hybrid approach in **D1**, the detection gains are smaller than those

#	Configuration	DSMV Backbone	Dataset **D1** (various vessel sizes) AP@0.2	AP@0.3	AP@0.4	Dataset **D2** (small vessels) AP@0.2	AP@0.3	AP@0.4
1	End-to-end: F-RCNN R-101 [1]	N/A	0.680	0.675	0.653	0.297	0.248	0.179
2	End-to-end: F-RCNN R-50 [2]	N/A	0.654	0.634	0.621	0.283	**0.256**	0.164
3	End-to-end: F-RCNN X-101 [3]	N/A	0.703	0.689	0.659	0.337	0.232	**0.186**
4	End-to-end: Cascade R-CNN R-50 [4]	N/A	0.699	0.680	0.662	0.271	0.229	0.151
5	End-to-end: RetinaNet R-101 [5]	N/A	**0.787**	**0.761**	**0.704**	**0.359**	0.240	0.134
6	Hybrid: F-RCNN R-101	ResNet-50	0.787	0.772	0.738	0.541	0.462	0.217
7	Hybrid: F-RCNN X-101	ResNet-50	0.774	0.756	0.720	0.570	0.457	0.295
8	Hybrid: Cascade R-CNN R-50	ResNet-50	0.798	0.771	0.736	0.553	0.487	0.242
9	Hybrid: RetinaNet R-101	ResNet-50	0.809	0.780	0.714	0.557	0.438	0.210
10	Hybrid: F-RCNN R-50	Inception-V3	0.750	0.730	0.700	0.445	0.408	0.241
11	Hybrid: F-RCNN X-101	Inception-V3	0.765	0.752	0.716	0.499	0.398	0.293
12	Hybrid: Cascade R-CNN R-50	Inception-V3	0.791	0.774	0.735	0.462	0.403	0.234
13	Hybrid: F-RCNN X-101	ResNext-101	0.785	0.767	0.729	**0.619**	0.506	**0.341**
14	Hybrid: Cascade R-CNN R-50	ResNext-101	0.814	0.787	**0.749**	0.608	**0.543**	0.282
15	Hybrid: RetinaNet R-101	ResNext-101	**0.833**	0.804	0.736	0.608	0.489	0.261
16	Hybrid: F-RCNN R-50	ResNext-50	0.747	0.726	0.701	0.464	0.420	0.224
17	Hybrid: F-RCNN X-101	ResNext-50	0.765	0.751	0.718	0.525	0.409	0.269
18	Hybrid: Cascade R-CNN R-50	ResNext-50	0.791	0.773	0.739	0.480	0.409	0.210
19	Hybrid: RetinaNet R-101	ResNext-50	0.826	0.800	0.736	0.514	0.389	0.201
20	Hybrid: F-RCNN X-101	Wide ResNet 50-2	0.779	0.761	0.723	**0.619**	0.506	**0.341**
21	Hybrid: Cascade R-CNN R-50	Wide ResNet 50-2	0.808	0.781	0.743	0.602	0.536	0.290
22	Hybrid: RetinaNet R-101	Wide ResNet 50-2	0.822	0.793	0.725	0.608	0.489	0.261
23	Hybrid: F-RCNN R-50	DenseNet-201	0.766	0.745	0.713	0.508	0.465	0.261
24	Hybrid: F-RCNN X-101	DenseNet-201	0.777	0.763	0.725	0.557	0.442	0.302
25	Hybrid: Cascade R-CNN R-50	DenseNet-201	0.804	0.786	0.747	0.526	0.457	0.251
26	Hybrid: RetinaNet R-101	DenseNet-201	0.832	**0.807**	0.738	0.533	0.408	0.224

[1,2,3]: Pre-trained Faster R-CNN [118] using FPN [297] with Resnet-101 [251], Resnet-50 [251] and ResNext-101 [303] as feature extractors, respectively.
[4,5]: Pre-trained Cascade R-CNN [282] and RetinaNet [281] using FPN [297] with Resnet-50 [251] and Resnet-101 [251] as feature extractors, respectively.

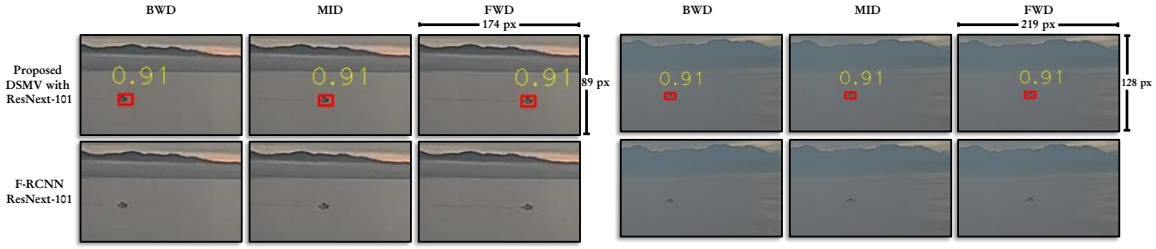
Table 5.2: Average Precision (AP) results for configurations combining pre-trained end-to-end object detectors, DSMV with custom-trained image classifiers, and different Intersection-over-Union thresholds (see 5.4.2 for details). Best results for each layout and dataset are highlighted in bold. Our hybrid approach outperforms all corresponding stand-alone end-to-end pre-trained object detectors.

observed in **D2** because, as mentioned, the end-to-end object detectors work well for detecting medium- and large-sized vessels.

The potential of the DSMV is more explicit when using considering dataset **D2**, where the improvement in average precision of the best-performing configurations for IoU threshold 0.3 (configurations #2 and #14) is 112.1%. On average, the boost in performance when considering the best configurations for all three IoU thresholds and **D2** is 89.28%. The average precision drops significantly on all configurations when the IoU threshold is increased because the small vessels of **D2** are hard to precisely encompass in either manual annotation or autonomous detection, as their visual boundaries are often blurry (see Figure 5.2).

Figure 5.11 illustrates detection results under different layouts. On the second row of Figure 5.11a, the output of Faster R-CNN using ResNext-101 show that the object detector missed all the small boats contained in these six excerpts from images of the **D2** dataset. The output from the proposed hybrid approach (first row of Figure 5.11a) highlights the ability of the DSMV to identify extremely small vessels

that were initially missed. Results for **D2** also show that the proposed detector is robust to non-horizontal movements (*i.e.*, assumption **A1**), given that some boats in it display a mostly concave trajectory. The results of the proposed system shown on Figure 5.11b distinguish between detection made by the DSMV (red bounding boxes) and otherwise (yellow bounding boxes). Note that the object detector (Faster R-CNN with ResNext-101) correctly identified medium- and large-sized boats, while most of the small-sized boats are identified only with the use of the DSMV.



(a) Detection results on dataset **D2** for end-to-end object detector (second row) and proposed hybrid layouts (first row). Red bounding boxes highlight DSMV detection.

(b) Detection results on dataset **D1** for the proposed hybrid approach (*i.e.*, pre-trained object detector output combined with the DSMV output) using Faster R-CNN and ResNext-101. Yellow bounding boxes indicate object detector-only results while red bounding boxes show the DSMV-generated output.

Figure 5.11: Detection results of our hybrid system and stand-alone object detectors (second row of (**a**)).

5.4.3 Implementation Details

We implemented the vessels detector system with PyTorch [4] using pre-trained weights and implementations from Detectron2 [5]. In our system the user can set an $x-$ and $y-$axis range of valid detection, allowing for regions of the image with static content (*e.g.*, manufacturer logo and date on Figure 5.11b) to be ignored during the detection

[4] https://pytorch.org
[5] https://github.com/facebookresearch/detectron2

process. The $y-$axis range set for **D1** and **D2** tests were, respectively, $[281, 850]$ and $[650, 896]$. Additionally, an $x-$axis range of $[132, 1920]$ is set on **D2** tests to ignore a ladder (see Figure 5.10b).

The vertical (temporal) band is delimited by a ρ_{vdb} of 2, while the horizontal (positional) band is set by a ψ_{hdb} of 1.4. Once a BMBB candidate is associated with a FMBB, its content is removed from further template matching tasks, thus the content of a BMBB can only be matched with a single FMBB. The MSE threshold (MSE_{th}) used in the experiments is 600. In order to avoid a large group of invalid candidates in the initial phases of the DSMV (*e.g.*, those often triggered by sunlight reflections), we limit the maximum number of FMBB considered. We start with a GMM threshold (squared Mahalanobis distance [305] between a pixel and the Gaussian distributions) of 120, and if the initial number of FMBB is higher than 18, we increase this threshold by 100 and calculate a new group of FMBB. Vessels typically create more pronounced deviations from background distributions, thus the progressively higher thresholds of this proposed algorithm eventually filter the invalid FMBB. We use Zivkovic's [306, 307] background subtraction method as implemented in OpenCV[6] as a basis for our bidirectional GMM. We prioritize the detection output (*i.e.*, bounding box dimensions, position and detection score) of the end-to-end object detectors when there exists an overlap with the output of the DSMV. Moreover, we set a standard detection score (for AP calculation purposes) for the DSMV of 0.91. Changing this value modifies the relevance assigned to DSMV detection, and marginally changes the overall AP associated to each configuration.

5.5 Conclusion

Our hybrid marine vessel detector uses short time series to identify boats of any size, shape, and under different viewing conditions. The proposed DSMV uses a combination of a novel bidirectional GMM strategy, classical Computer Vision methods and custom-trained DL-based classifiers for identifying challenging small vessels. Extensive experiments show that our hybrid approach outperforms five state-of-the-art object detectors on two datasets we make publicly available.

The proposed detector fulfills real-world automated processing needs of data managers and governance [308], in particular in critical habitats such as the Salish Sea. Its

[6] www.opencv.org

fast (approximately 0.4 seconds per image) and efficient detection enables the timely interpretation of visual Environmental Monitoring data to support conservation and research efforts. We also provide novel visual datasets of AIS and non-AIS vessel fleets in important ecological areas to promote further research.

Our approach is based on a set of four assumptions that might not be representative of all monitoring layouts, thus one must be mindful of them before employing our proposed system. Small adaptations (*e.g.*, camera tilting, image pre-processing) can assist in ensuring that these assumptions are valid for the visual EM data being processed. Different image frame rates can be employed by enlarging the time window considered by the bidirectional GMM. Boats that move towards the camera or are partially occluded (*e.g.*, by other vessels) might result in false negatives from the DSMV.

Future work will involve ablation studies and the use of different background modelling strategies (*e.g.*, Bloisi *et al.* discrete distribution [283]) in the first stages of the DSMV. Other methods to encode temporal information (*e.g.*, Long Short-Term Memory (LSTM) networks [309]) and object tracking strategies [310] are also going to be considered.

5.6 Acknowledgments

The authors wish to acknowledge funding from Canada's Department of Fisheries and Oceans in support of Tunai Porto Marques, and the in-kind support from the Saturna Island Marine Research and Education Society, and Eagle Cove Beachfront Guest Suites for hosting our camera systems.

Chapter 6

Conclusion

The main goal of this thesis was to explore the capabilities, requirements, limitations, use cases and overall considerations when employing Computer Vision (CV) towards the interpretation of visual Environmental Monitoring (EM) data. This objective is summarized by the following research question (first presented in Chapter 1):

> *Can Computer Vision techniques be broadly employed to efficiently and accurately process years-worth of Environmental Monitoring visual data, given their unique natures, challenges, and scientific considerations?*

In order to decompose this broad analysis into feasible research units, we elected a non-exhaustive list of three types of EM data streams to serve as representatives of the output from diverse monitoring layouts: underwater visual, out-of-water visual and underwater active acoustic (see Figure 1.6).

While thoroughly exploring the use of Computer Vision for the automatic interpretation of each stream, we also studied, proposed and employed a host of approaches that typically permeate this research field: "traditional" Computer Vision and Machine Learning (both DL-based or otherwise). The projects completed in this thesis led to two broad observations: 1) the three aforementioned methodologies are all valuable under diverse circumstances dictated by the data available, and 2) the choice of CV approach to be used in EM is highly application-dependent. Chapter 2 describes a framework designed around physical image formation models, illustrating a scenario where "traditional" CV techniques are appropriate, given their ability to directly implement contextual observations (*e.g.*, contrast-based calculation of patch sizes, lighting and transmission maps). Chapter 3 describes a system that is completely based

on the feature-extraction ability of a DL method—highlighting that non-trivial morphological structures can be effectively identified (see sub-section 3.4) using CNNs. Conversely, Chapter 4 provides a comparison study that shows that the "manual" (*i.e.*, without the use of deep neural networks) extraction of visual features, combined with the use of traditional ML (decision trees [233], Naive Bayes [234], SVM [235], KNN [236]) also yields powerful detection capabilities, often over-performing some DL-based systems (see Table 4.2). Finally, Chapter 5 presents an EM scenario where the use of either DL or "traditional" CV alone did not suffice. The hybrid vessel detection framework proposed employed DL-based object detectors and combined a number of "traditional" CV-based steps to manually account for environmental assumptions, ultimately identifying boats under various appearances.

The results and discussions presented in the last four Chapters serve as evidence and allow responding our main research question: in fact, Computer Vision represents a powerful family of techniques and capabilities for the interpretation of visual EM data. By designing an efficient CV-based system, often specific to an application, one might address the multiple challenges and requirements (*e.g.*, volume, performance, consistency, structure, accuracy) associated with EM studies.

Volume, performance and consistency: while the sheer volume of data that is typically acquired in EM initiatives often prevents their manual interpretation, well-designed CV-based systems can process these quickly and consistently. For instance, the marine vessel detection system proposed in Chapter 5 has already been used to process an excess of 250 thousand visual EM datapoints. Considering a conservative scenario where a human operator would efficiently process each sample in 3 seconds, this dataset would only be completely analyzed in almost 9 days of uninterrupted work. In contrast, the detection system proposed in Chapter 5 takes approximately 0.4 seconds to identify marine vessels on a sample, thus processing 250 thousand samples in approximately 28 hours. Moreover, the accurate (see sub-section 5.4.2) output of the detector proposed is consistent (*i.e.*, a given input will always lead to the same output) and does not suffer from fatigue, human errors, biases, or any other aggravating circumstances. Similarly, the U-MSAA-Net framework proposed in Chapter 4 processes a pair of multi-frequency echograms efficiently (see Table 4.2) in merely 0.02 seconds.

Accuracy: the manually-created ground truths used to evaluate the performance of the systems proposed in Chapters 3, 4 and 5 represent expert-led interpretations of these EM data. Therefore, the high levels of accuracy observed in the proposed

systems (refer to the Results sections of the last three Chapters) attest to their ability to reproduce the interpretation capabilities of an expert in the field. The detector of schools of salmon and herring proposed in Chapter 3, for example, achieved a mean Average Precision of 0.921 for IoU thresholds of 0.5. As illustrated in Figure 3.4, these output reached a level of accuracy that is comparable to that of human operators (*i.e.*, similar to the ground-truth). U-MSAA-Net achieved IoU, precision and recall metrics of more than 0.96 in both 67 and 125 kHz echograms (Table 4.2), significantly closing the gap between CV-based and expert-led interpretations of acoustic EM data.

Structure: while some goals of EM studies do not require high levels of expertise from human operators (*e.g.*, "identify marine vessels in monitoring images"), other goals call for niche technical knowledge (*e.g.*, "identify schools of herring, juvenile salmon and hake in acoustic backscatter signals"). Assuming that enough expert annotations are available, CV systems can directly learn from examples and thus implicitly find patterns that allow them to efficiently interpret these data. Furthermore, non-data-driven CV systems might also perform tasks related to visual EM data would be difficult to complete manually (*e.g.*, "reveal the visual content hidden by regions of low visibility in low-light underwater images"). The image enhancement system proposed in Chapter 2, for instance, uses assumptions about the physical formation of hazy images (Equation 2.1) to recover pixel intensities mitigated by environmental conditions—a challenging task to carry out manually and that can greatly boost the scientific value of underwater visual EM data.

Further work on the projects composing this thesis might involve the use of additional visual data (*e.g.*, infrared imagery), as well as a detailed study of inherently challenging streams (*e.g.*, monitoring *videos*). For monitoring videos, Convolutional Neural Networks applied simultaneously to multiple frames (*e.g.*, [311, 74]) could encode valuable temporal relationships, allowing for robust analyses of their content. Studies that compare the performances of a CV-based system against that of human operators are also recommended for an evaluation that focuses on the benefits that the proposed systems bring to real-world applications. In terms of technologies, a deeper exploration of recent visual frameworks such as the attention-based Transformer [312] (similarly to the MSAA module debuted in Chapter 4) is recommended. More specifically, approaches based on attention mechanisms have been used towards a more holistic interpretation of images [313, 314, 315, 316], and could potentially increase the performance of the CNN-based frameworks proposed in this thesis and other EM applications.

Bibliography

[1] J. F. Artiola, M. L. Brusseau, and I. L. Pepper, *Environmental monitoring and characterization*. Academic Press, 2004.

[2] C. Parmesan, "Ecological and evolutionary responses to recent climate change," *Annu. Rev. Ecol. Evol. Syst.*, vol. 37, pp. 637–669, 2006.

[3] C. Mora, A. G. Frazier, R. J. Longman, R. S. Dacks, M. M. Walton, E. J. Tong, J. J. Sanchez, L. R. Kaiser, Y. O. Stender, J. M. Anderson, *et al.*, "The projected timing of climate departure from recent variability," *Nature*, vol. 502, no. 7470, p. 183, 2013.

[4] S. J. Cooke and J. F. Schreer, "Environmental monitoring using physiological telemetry–a case study examining common carp responses to thermal pollution in a coal-fired generating station effluent," *Water, air, and soil pollution*, vol. 142, no. 1, pp. 113–136, 2003.

[5] G. M. Lovett, D. A. Burns, C. T. Driscoll, J. C. Jenkins, M. J. Mitchell, L. Rustad, J. B. Shanley, G. E. Likens, and R. Haeuber, "Who needs environmental monitoring?," *Frontiers in Ecology and the Environment*, vol. 5, no. 5, pp. 253–260, 2007.

[6] G. Houtven, S. Brunnermeier, and M. Buckley, "A retrospective assessment of the costs of the clean water act: 1972 to 1997," 2000.

[7] J. M. Johnson, "The cost of regulations implementing the clean water act: a working paper in regulatory studies," 2004.

[8] L. G. Chestnut and D. M. Mills, "A fresh look at the benefits and costs of the us acid rain program," *Journal of Environmental Management*, vol. 77, no. 3, pp. 252–266, 2005.

[9] E. S. Bernhardt, M. A. Palmer, J. Allan, G. Alexander, K. Barnas, S. Brooks, J. Carr, S. Clayton, C. Dahm, J. Follstad-Shah, *et al.*, "Synthesizing us river restoration efforts," 2005.

[10] P. Spachos, L. Song, and D. Hatzinakos, "Prototypes of opportunistic wireless sensor networks supporting indoor air quality monitoring," in *2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*, pp. 851–852, IEEE, 2013.

[11] N. Kotamäki, S. Thessler, J. Koskiaho, A. Hannukkala, H. Huitu, T. Huttula, J. Havento, and M. Järvenpää, "Wireless in-situ sensor network for agriculture and water monitoring on a river basin scale in southern finland: Evaluation from a data user's perspective," *Sensors*, vol. 9, no. 4, pp. 2862–2883, 2009.

[12] P. Jiang, H. Xia, Z. He, and Z. Wang, "Design of a water environment monitoring system based on wireless sensor networks," *Sensors*, vol. 9, no. 8, pp. 6411–6434, 2009.

[13] B. O'Flynn, R. Martinez-Catala, S. Harte, C. O'Mathuna, J. Cleary, C. Slater, F. Regan, D. Diamond, and H. Murphy, "Smartcoast: a wireless sensor network for water quality monitoring," in *32nd IEEE Conference on Local Computer Networks (LCN 2007)*, pp. 815–816, Ieee, 2007.

[14] A. S. Lutakamale and S. Kaijage, "Wildfire monitoring and detection system using wireless sensor network: A case study of tanzania," *Wireless Sensor Network (WSN)*, vol. 9, no. 8, 2017.

[15] M. Trincavelli, M. Reggente, S. Coradeschi, A. Loutfi, H. Ishida, and A. J. Lilienthal, "Towards environmental monitoring with mobile robots," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2210–2215, IEEE, 2008.

[16] P. Weibring, H. Edner, and S. Svanberg, "Versatile mobile lidar system for environmental monitoring," *Applied Optics*, vol. 42, no. 18, pp. 3583–3594, 2003.

[17] R. Hes, F. Kempf, J. Tautz, and K. Schilling, "Remote biological and robotic sensor networks for environmental monitoring," *IFAC Proceedings Volumes*, vol. 46, no. 29, pp. 138–143, 2013.

[18] M. C. Hansen and T. R. Loveland, "A review of large area monitoring of land cover change using landsat data," *Remote sensing of Environment*, vol. 122, pp. 66–74, 2012.

[19] F. Nex and F. Remondino, "Uav for 3d mapping applications: a review," *Applied geomatics*, vol. 6, no. 1, pp. 1–15, 2014.

[20] M. A. Nieto, B. Garau, S. Balle, G. Simarro, G. A. Zarruk, A. Ortiz, J. Tintoré, A. Álvarez-Ellacuría, L. Gómez-Pujol, and A. Orfila, "An open source, low cost video-based coastal monitoring system," *Earth Surface Processes and Landforms*, vol. 35, no. 14, pp. 1712–1719, 2010.

[21] W. Ding and G. Taylor, "Automatic moth detection from trap images for pest management," *Computers and Electronics in Agriculture*, vol. 123, pp. 17–28, 2016.

[22] M. Galaverni, D. Palumbo, E. Fabbri, R. Caniglia, C. Greco, and E. Randi, "Monitoring wolves (canis lupus) by non-invasive genetics and camera trapping: a small-scale pilot study," *European Journal of Wildlife Research*, vol. 58, no. 1, pp. 47–58, 2012.

[23] M. S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, M. S. Palmer, C. Packer, and J. Clune, "Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning," *Proceedings of the National Academy of Sciences*, vol. 115, no. 25, pp. E5716–E5725, 2018.

[24] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, "Automatic fish classification for underwater species behavior understanding," in *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, pp. 45–50, ACM, 2010.

[25] P. Fearns, W. Klonowski, R. Babcock, P. England, and J. Phillips, "Shallow water substrate mapping using hyperspectral remote sensing," *Continental Shelf Research*, vol. 31, no. 12, pp. 1249–1259, 2011.

[26] J. C. Baker and R. A. Williamson, "Satellite imagery activism: sharpening the focus on tropical deforestation," *Singapore Journal of Tropical Geography*, vol. 27, no. 1, pp. 4–14, 2006.

[27] N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov, "Deep learning classification of land cover and crop types using remote sensing data," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 5, pp. 778–782, 2017.

[28] M. J. Leotta, C. Long, B. Jacquet, M. Zins, D. Lipsa, J. Shan, B. Xu, Z. Li, X. Zhang, S.-F. Chang, *et al.*, "Urban semantic 3d reconstruction from multiview satellite imagery," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 0–0, 2019.

[29] P. T. Fretwell, M. A. LaRue, P. Morin, G. L. Kooyman, B. Wienecke, N. Ratcliffe, A. J. Fox, A. H. Fleming, C. Porter, and P. N. Trathan, "An emperor penguin population estimate: the first global, synoptic survey of a species from space," *PloS one*, vol. 7, no. 4, p. e33751, 2012.

[30] D. A. Contreras and N. Brodie, "The utility of publicly-available satellite imagery for investigating looting of archaeological sites in jordan," *Journal of Field Archaeology*, vol. 35, no. 1, pp. 101–114, 2010.

[31] K. Briess, H. Jahn, E. Lorenz, D. Oertel, W. Skrbek, and B. Zhukov, "Fire recognition potential of the bi-spectral infrared detection (bird) satellite," *International Journal of Remote Sensing*, vol. 24, no. 4, pp. 865–872, 2003.

[32] M. Vittek, A. Brink, F. Donnay, D. Simonetti, and B. Desclée, "Land cover change monitoring using landsat mss/tm satellite image data over west africa between 1975 and 1990," *Remote Sensing*, vol. 6, no. 1, pp. 658–676, 2014.

[33] P. V. Potapov, S. A. Turubanova, M. C. Hansen, B. Adusei, M. Broich, A. Altstatt, L. Mane, and C. O. Justice, "Quantifying forest cover loss in democratic republic of the congo, 2000–2010, with landsat etm+ data," *Remote Sensing of Environment*, vol. 122, pp. 106–116, 2012.

[34] B. Johansen and S. R. Karlsen, "Monitoring vegetation changes on finnmarksvidda, northern norway, using landsat mss and landsat tm/etm+ satellite images," *Phytocoenologia*, vol. 35, no. 4, pp. 969–984, 2005.

[35] J. M. Chen and J. Cihlar, "Retrieving leaf area index of boreal conifer forests using landsat tm images," *Remote sensing of Environment*, vol. 55, no. 2, pp. 153–162, 1996.

[36] S. Kilic, F. Evrendilek, S. Berberoglu, and A. Demirkesen, "Environmental monitoring of land-use and land-cover changes in a mediterranean region of turkey," *Environmental monitoring and assessment*, vol. 114, no. 1-3, pp. 157–168, 2006.

[37] T. Rosnell and E. Honkavaara, "Point cloud generation from aerial image data acquired by a quadrocopter type micro unmanned aerial vehicle and a digital still camera," *Sensors*, vol. 12, no. 1, pp. 453–480, 2012.

[38] J. P. Dandois and E. C. Ellis, "High spatial resolution three-dimensional mapping of vegetation spectral dynamics using computer vision," *Remote Sensing of Environment*, vol. 136, pp. 259–276, 2013.

[39] H. Bischof, W. Schneider, and A. J. Pinz, "Multispectral classification of landsat-images using neural networks," *IEEE transactions on Geoscience and Remote Sensing*, vol. 30, no. 3, pp. 482–490, 1992.

[40] C. Wu, F. Fraundorfer, J.-M. Frahm, J. Snoeyink, and M. Pollefeys, "Image localization in satellite imagery with feature-based indexing," *Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Beijing: ISPRS*, vol. 37, pp. 197–202, 2008.

[41] J. Gonçalves and R. Henriques, "Uav photogrammetry for topographic monitoring of coastal areas," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 104, pp. 101–111, 2015.

[42] H. Aasen, A. Burkart, A. Bolten, and G. Bareth, "Generating 3d hyperspectral information with lightweight uav snapshot cameras for vegetation monitoring: From camera calibration to quality assurance," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 108, pp. 245–259, 2015.

[43] H. Bendea, P. Boccardo, S. Dequal, F. G. Tonolo, D. Marenchino, and M. Piras, "Low cost uav for post-disaster assessment," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 37, no. B8, pp. 1373–1379, 2008.

[44] M. Aljehani and M. Inoue, "Performance evaluation of multi-uav system in post-disaster application: Validated by hitl simulator," *IEEE Access*, vol. 7, pp. 64386–64400, 2019.

[45] C. A. F. Ezequiel, M. Cua, N. C. Libatique, G. L. Tangonan, R. Alampay, R. T. Labuguen, C. M. Favila, J. L. E. Honrado, V. Canos, C. Devaney, *et al.*, "Uav aerial imaging applications for post-disaster assessment, environmental management and infrastructure development," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 274–283, IEEE, 2014.

[46] J. Bendig, K. Yu, H. Aasen, A. Bolten, S. Bennertz, J. Broscheit, M. L. Gnyp, and G. Bareth, "Combining uav-based plant height from crop surface models, visible, and near infrared vegetation indices for biomass monitoring in barley," *International Journal of Applied Earth Observation and Geoinformation*, vol. 39, pp. 79–87, 2015.

[47] B. Kellenberger, D. Marcos, and D. Tuia, "Detecting mammals in uav images: Best practices to address a substantially imbalanced dataset with deep learning," *Remote sensing of environment*, vol. 216, pp. 139–153, 2018.

[48] J. C. Hodgson, S. M. Baylis, R. Mott, A. Herrod, and R. H. Clarke, "Precision wildlife monitoring using unmanned aerial vehicles," *Scientific reports*, vol. 6, p. 22574, 2016.

[49] Z. Zhu, A. R. Hanson, H. Schultz, F. Stolle, and E. M. Riseman, "Stereo mosaics from a moving video camera for environmental monitoring," in *First international workshop on digital and computational video*, pp. 45–54, 1999.

[50] N. Haala, "Comeback of digital image matching," in *Photogrammetric Week*, vol. 9, pp. 289–301, Ed. D. Fritsch Heidelberg, 2009.

[51] A. S. Laliberte, J. E. Herrick, A. Rango, and C. Winters, "Acquisition, orthorectification, and object-based classification of unmanned aerial vehicle (uav) imagery for rangeland monitoring," *Photogrammetric Engineering & Remote Sensing*, vol. 76, no. 6, pp. 661–672, 2010.

[52] S. Siebert and J. Teizer, "Mobile 3d mapping for surveying earthwork projects using an unmanned aerial vehicle (uav) system," *Automation in construction*, vol. 41, pp. 1–14, 2014.

[53] A. Lingua, D. Marenchino, and F. Nex, "Performance analysis of the sift operator for automatic feature extraction and matching in photogrammetric applications," *Sensors*, vol. 9, no. 5, pp. 3745–3766, 2009.

[54] R. Holman, J. Stanley, and T. Ozkan-Haller, "Applying video sensor networks to nearshore environment monitoring," *IEEE Pervasive Computing*, vol. 2, no. 4, pp. 14–21, 2003.

[55] B. Hobley, R. Arosio, G. French, J. Bremner, T. Dolphin, and M. Mackiewicz, "Semi-supervised segmentation for coastal monitoring seagrass using rpa imagery," *Remote Sensing*, vol. 13, no. 9, p. 1741, 2021.

[56] T. P. Marques, A. B. Albu, P. O'Hara, N. Serra, B. Morrow, L. McWhinnie, and R. Canessa, "Size-invariant detection of marine vessels from visual time series," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 443–453, 2021.

[57] A. Papakonstantinou, M. Batsaris, S. Spondylidis, and K. Topouzelis, "A citizen science unmanned aerial system data acquisition protocol and deep learning techniques for the automatic detection and mapping of marine litter concentrations in the coastal zone," *Drones*, vol. 5, no. 1, p. 6, 2021.

[58] R. A. Holman and J. Stanley, "The history and technical capabilities of argus," *Coastal engineering*, vol. 54, no. 6-7, pp. 477–491, 2007.

[59] M. D. Harley, I. L. Turner, A. D. Short, and R. Ranasinghe, "Assessment and integration of conventional, rtk-gps and image-derived beach survey methods for daily to decadal coastal monitoring," *Coastal Engineering*, vol. 58, no. 2, pp. 194–205, 2011.

[60] F. Baselice and G. Ferraioli, "Unsupervised coastal line extraction from sar images," *IEEE Geoscience and Remote Sensing Letters*, vol. 10, no. 6, pp. 1350–1354, 2013.

[61] N. Long, B. Millescamps, B. Guillot, F. Pouget, and X. Bertin, "Monitoring the topography of a dynamic tidal inlet using uav imagery," *Remote Sensing*, vol. 8, no. 5, p. 387, 2016.

[62] J. Graber, J. Thomson, B. Polagye, and A. Jessup, "Land-based infrared imagery for marine mammal detection," in *Remote Sensing and Modeling of Ecosystems for Sustainability VIII*, vol. 8156, p. 81560B, International Society for Optics and Photonics, 2011.

[63] V. Santhaseelan and V. K. Asari, "Automated whale blow detection in infrared video," in *Computer Vision and Pattern Recognition in Environmental Informatics*, pp. 58–78, IGI Global, 2016.

[64] V. Anton, S. Hartley, A. Geldenhuis, and H. U. Wittmer, "Monitoring the mammalian fauna of urban areas using remote cameras and citizen science," *Journal of Urban Ecology*, vol. 4, no. 1, p. juy002, 2018.

[65] A. G. Villa, A. Salazar, and F. Vargas, "Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks," *Ecological Informatics*, vol. 41, pp. 24–32, 2017.

[66] X. Yu, J. Wang, R. Kays, P. A. Jansen, T. Wang, and T. Huang, "Automated identification of animal species in camera trap images," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, p. 52, 2013.

[67] S. B. Williams, O. R. Pizarro, M. V. Jakuba, C. R. Johnson, N. S. Barrett, R. C. Babcock, G. A. Kendrick, P. D. Steinberg, A. J. Heyward, P. J. Doherty, *et al.*, "Monitoring of benthic reference sites: using an autonomous underwater vehicle," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 73–84, 2012.

[68] I. Dumke, S. M. Nornes, A. Purser, Y. Marcon, M. Ludvigsen, S. L. Ellefmo, G. Johnsen, and F. Søreide, "First hyperspectral imaging survey of the deep seafloor: High-resolution mapping of manganese nodules," *Remote sensing of environment*, vol. 209, pp. 19–30, 2018.

[69] T. P. Marques and A. B. Albu, "L2uwe: A framework for the efficient enhancement of low-light underwater images using local contrast and multi-scale fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 538–539, 2020.

[70] T. Porto Marques, A. Branzan Albu, and M. Hoeberechts, "A contrast-guided approach for the enhancement of low-lighting underwater images," *MDPI Journal of Imaging*, vol. 5, no. 10, p. 79, 2019.

[71] C. O. Ancuti, C. Ancuti, C. De Vleeschouwer, and P. Bekaert, "Color balance and fusion for underwater image enhancement," *IEEE Transactions on image processing*, vol. 27, no. 1, pp. 379–393, 2017.

[72] C. Ancuti, C. O. Ancuti, T. Haber, and P. Bekaert, "Enhancing underwater images and videos by fusion," in *2012 IEEE conference on computer vision and pattern recognition*, pp. 81–88, IEEE, 2012.

[73] A. Salman, A. Jalal, F. Shafait, A. Mian, M. Shortis, J. Seager, and E. Harvey, "Fish species classification in unconstrained underwater environments based on deep learning," *Limnology and Oceanography: Methods*, vol. 14, no. 9, pp. 570–585, 2016.

[74] D. McIntosh, T. P. Marques, A. B. Albu, R. Rountree, and F. De Leo, "Movement tracks for the automatic detection of fish behavior in videos," *arXiv preprint arXiv:2011.14070*, 2020.

[75] M. Ford, N. Bezio, and A. Collins, "Duobrachium sparksae (incertae sedis ctenophora tentaculata cydippida): A new genus and species of benthopelagic ctenophore seen at 3,910 m depth off the coast of puerto rico," *Plankton and Benthos Research*, vol. 15, no. 4, pp. 296–305, 2020.

[76] T. P. Marques, M. Cote, A. Rezvanifar, A. B. Albu, K. Ersahin, T. Mudge, and S. Gauthier, "Instance segmentation-based identification of pelagic species in acoustic backscatter data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4378–4387, June 2021.

[77] T. P. Marques, A. Rezvanifar, M. Cote, A. B. Albu, K. Ersahin, T. Mudge, and S. Gauthier, "Detecting marine species in echograms via traditional, hybrid, and deep learning frameworks," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 5928–5935, IEEE, 2021.

[78] A. Rezvanifar, T. P. Marques, M. Cote, A. B. Albu, A. Slonimer, T. Tolhurst, K. Ersahin, T. Mudge, and S. Gauthier, "A deep learning-based framework for the detection of schools of herring in echograms," *arXiv preprint arXiv:1910.08215*, 2019.

[79] A. Gleason, R. Reid, and K. Voss, "Automated classification of underwater multispectral imagery for coral reef monitoring," in *OCEANS 2007*, pp. 1–8, IEEE, 2007.

[80] A. Friedman, O. Pizarro, S. B. Williams, and M. Johnson-Roberson, "Multi-scale measures of rugosity, slope and aspect from benthic stereo image reconstructions," *PloS one*, vol. 7, no. 12, p. e50440, 2012.

[81] R. B. Fisher, K.-T. Shao, and Y.-H. Chen-Burger, "Overview of the fish4knowledge project," in *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*, pp. 1–17, Springer, 2016.

[82] A. G. Cabreira, M. Tripode, and A. Madirolas, "Artificial neural networks for fish-species identification," *ICES Journal of Marine Science*, vol. 66, no. 6, pp. 1119–1129, 2009.

[83] S. A. Villar, A. Madirolas, A. G. Cabreira, A. Rozenfeld, and G. G. Acosta, "Ecopampa: A new tool for automatic fish schools detection and assessment from echo data," *Heliyon*, vol. 7, no. 1, p. e05906, 2021.

[84] H. Robotham, P. Bosch, J. C. Gutiérrez-Estrada, J. Castillo, and I. Pulido-Calvo, "Acoustic identification of small pelagic fish species in Chile using support vector machines and neural networks," *Fisheries Research*, vol. 102, no. 1-2, pp. 115–122, 2010.

[85] N. G. Fallon, S. Fielding, and P. G. Fernandes, "Classification of Southern Ocean krill and icefish echoes using random forests," *ICES Journal of Marine Science*, vol. 73, no. 8, pp. 1998–2008, 2016.

[86] Y. Hirama, S. Yokoyama, T. Yamashita, H. Kawamura, K. Suzuki, and M. Wada, "Discriminating fish species by an Echo sounder in a set-net using a CNN," in *21st Asia Pacific Symposium on Intelligent and Evolutionary Systems (IES)*, pp. 112–115, IEEE, 2017.

[87] O. Brautaset, A. U. Waldeland, E. Johnsen, K. Malde, L. Eikvil, A.-B. Salberg, *et al.*, "Acoustic classification in multifrequency echosounder data using deep convolutional neural networks," *ICES Journal of Marine Science*, 2020.

[88] M. A. Wulder, J. C. White, T. R. Loveland, C. E. Woodcock, A. S. Belward, W. B. Cohen, E. A. Fosnight, J. Shaw, J. G. Masek, and D. P. Roy, "The global landsat archive: Status, consolidation, and direction," *Remote Sensing of Environment*, vol. 185, pp. 271–283, 2016.

[89] A. Swanson, M. Kosmala, C. Lintott, R. Simpson, A. Smith, and C. Packer, "Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna," 2015.

[90] M. Kosmala, A. Wiggins, A. Swanson, and B. Simmons, "Assessing data quality in citizen science," *Frontiers in Ecology and the Environment*, vol. 14, no. 10, pp. 551–560, 2016.

[91] A. Swanson, T. Arnold, M. Kosmala, J. Forester, and C. Packer, "In the absence of a "landscape of fear": How lions, hyenas, and cheetahs coexist," *Ecology and evolution*, vol. 6, no. 23, pp. 8534–8545, 2016.

[92] J. Roff and M. Zacharias, *Marine Conservation Ecology*, ch. 1, pp. 1–8. Routledge, 1 ed., 6 2011.

[93] J. L. Salle, *Computer vision and pattern recognition in environmental informatics: Foreword*. IGI Global, 2015.

[94] L. Shapiro and G. Stockman, *Computer Vision*, ch. 1, p. 14. Prentice-Hall, 2001.

[95] S. Hagen, "The mind's eye," *Rochester Review*, vol. 74, no. 4, pp. 32–37, 2012.

[96] K. Novak, "Rectification of digital imagery," *Photogrammetric engineering and remote sensing*, vol. 58, pp. 339–339, 1992.

[97] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. The Edinburgh Building, Cambridge CB2 2RU, UK: Cambridge University Press, 2 ed., 2003.

[98] R. C. Gonzales and R. E. Woods, *Digital image processing using MATLAB*, ch. 2, pp. 47–55. McGraw Hill Education, 2010.

[99] U. S. G. S. (USGS), "Using the usgs landsat level-1 data product."

[100] P. S. Chavez Jr, "An improved dark-object subtraction technique for atmospheric scattering correction of multispectral data," *Remote sensing of environment*, vol. 24, no. 3, pp. 459–479, 1988.

[101] T. P. Marques, A. B. Albu, and M. Hoeberechts, "Enhancement of low-lighting underwater images using dark channel prior and fast guided filters," in *ICPR 3rd Workshop on Computer Vision for Analysis of Underwater Imagery (CVAUI)*, IAPR, 2018.

[102] K. Zuiderveld, "Contrast limited adaptive histogram equalization," *Graphics gems*, pp. 474–485, 1994.

[103] R. Hummel, "Image enhancement by histogram transformation," *Computer Graphics and Image Processing*, vol. 6, 1977.

[104] M. Wulder, R. Skakun, W. Kurz, and J. White, "Estimating time since forest harvest using segmented landsat etm+ imagery," *Remote sensing of environment*, vol. 93, no. 1-2, pp. 179–187, 2004.

[105] G. Padmavathi, M. Muthukumar, and S. K. Thakur, "Non linear image segmentation using fuzzy c means clustering method with thresholding for underwater images," *International Journal of Computer Science Issues (IJCSI)*, vol. 7, no. 3, p. 35, 2010.

[106] D. Akkaynak and T. Treibitz, "Sea-thru: A method for removing water from underwater images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1682–1691, 2019.

[107] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679–698, 1986.

[108] J. M. Prewitt, "Object enhancement and extraction," *Picture processing and Psychopictorics*, vol. 10, no. 1, pp. 15–19, 1970.

[109] L. G. Roberts, *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1963.

[110] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[111] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *International journal of computer vision*, vol. 74, no. 1, pp. 59–73, 2007.

[112] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*, pp. 404–417, Springer, 2006.

[113] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*, pp. 778–792, Springer, 2010.

[114] S. Leutenegger, M. Chli, and R. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 IEEE international conference on computer vision (ICCV)*, pp. 2548–2555, Ieee, 2011.

[115] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "Orb: An efficient alternative to sift or surf.," in *ICCV*, vol. 11, p. 2, Citeseer, 2011.

[116] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: Learned invariant feature transform," in *European conference on computer vision*, pp. 467–483, Springer, 2016.

[117] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[118] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, pp. 91–99, 2015.

[119] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, Springer, 2015.

[120] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2961–2969, 2017.

[121] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge, "Deeplabcut: markerless pose estimation of user-defined body parts with deep learning," *Nature neuroscience*, vol. 21, no. 9, pp. 1281–1289, 2018.

[122] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3d object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 5, pp. 1578–1604, 2019.

[123] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[124] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European conference on computer vision*, pp. 649–666, Springer, 2016.

[125] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[126] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[127] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[128] A. Baldacci, M. J. Carron, and N. Portunato, "Infrared detection of marine mammals."

[129] D. Lemon, P. Johnston, J. Buermans, E. Loos, G. Borstad, and L. Brown, "Multiple-frequency moored sonar for continuous observations of zooplankton and fish," in *2012 Oceans*, pp. 1–6, IEEE, 2012.

[130] X. Dong, G. Wang, Y. Pang, W. Li, J. Wen, W. Meng, and Y. Lu, "Fast efficient algorithm for enhancement of low lighting video," in *2011 IEEE International Conference on Multimedia and Expo*, IEEE, 2011.

[131] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.

[132] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE transactions on pattern analysis & machine intelligence*, vol. 35, no. 6, pp. 1397–1409, 2013.

[133] C. O. Ancuti and C. Ancuti, "Single image dehazing by multi-scale fusion," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3271–3282, 2013.

[134] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, 2017.

[135] A. Slonimer, M. Cote, T. P. Marques, A. Rezvanifar, S. Dosso, A. B. Albu, K. Ersahin, T. Mudge, and S. Gauthier, "Instance segmentation of herring and salmon schools in acoustic echograms using a hybrid u-net," in *2022 19th Conference on Robots and Vision (CRV)*, IEEE, 2022.

[136] D. McIntosh, T. P. Marques, A. B. Albu, R. Rountree, and F. De Leo, "Tempnet: Temporal attention towards the detection of animal behaviour in videos," in *2022 26th International Conference on Pattern Recognition (ICPR)*, IEEE, 2022.

[137] D. McIntosh, T. P. Marques, and A. B. Albu, "Preservation of high frequency content for deep learning-based medical image classification," in *2021 18th Conference on Robots and Vision (CRV)*, pp. 41–48, IEEE, 2021.

[138] T. Porto Marques, A. Branzan Albu, and M. Hoeberechts, "A contrast-guided approach for the enhancement of low-lighting underwater images," *Journal of Imaging*, vol. 5, no. 10, p. 79, 2019.

[139] D. Mallet and D. Pelletier, "Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952–2012)," *Fisheries Research*, vol. 154, pp. 44–62, 2014.

[140] D. Berman, T. Treibitz, and S. Avidan, "Diving into hazelines: Color restoration of underwater images," in *Proceedings of the British Machine Vision Conference (BMVC)*, vol. 1, 2017.

[141] Y. Cho and A. Kim, "Visibility enhancement for underwater visual slam based on underwater light scattering model," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 710–717, IEEE, 2017.

[142] P. Drews, E. Nascimento, F. Moraes, S. Botelho, and M. Campos, "Transmission estimation in underwater single images," in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 825–830, 2013.

[143] X. Fu, P. Zhuang, Y. Huang, Y. Liao, X.-P. Zhang, and X. Ding, "A retinex-based enhancing approach for single underwater image," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 4572–4576, IEEE, 2014.

[144] X. Guo, Y. Li, and H. Ling, "Lime: Low-light image enhancement via illumination map estimation," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 982–993, 2016.

[145] J. Zhang, Y. Cao, S. Fang, Y. Kang, and C. Wen Chen, "Fast haze removal for nighttime image using maximum reflectance prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7418–7426, 2017.

[146] K. Panetta, C. Gao, and S. Agaian, "Human-visual-system-inspired underwater image quality measures," *IEEE Journal of Oceanic Engineering*, vol. 41, no. 3, pp. 541–551, 2015.

[147] S. Wang, K. Ma, H. Yeganeh, Z. Wang, and W. Lin, "A patch-structure representation method for quality assessment of contrast changed images," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2387–2390, 2015.

[148] L. K. Choi, J. You, and A. C. Bovik, "Referenceless prediction of perceptual fog density and perceptual image defogging," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3888–3901, 2015.

[149] K. Matkovic, L. Neumann, A. Neumann, T. Psik, and W. Purgathofer, "Global contrast factor-a new approach to image contrast.," *Computational Aesthetics*, vol. 2005, pp. 159–168, 2005.

[150] N. Hautiere, J.-P. Tarel, D. Aubert, and E. Dumont, "Blind contrast enhancement assessment by gradient ratioing at visible edges," *Image Analysis & Stereology*, vol. 27, no. 2, pp. 87–95, 2011.

[151] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[152] M. Chambah, D. Semani, A. Renouf, P. Courtellemont, and A. Rizzi, "Underwater color constancy: enhancement of automatic live fish recognition," in *Color Imaging IX: Processing, Hardcopy, and Applications*, vol. 5293, pp. 157–169, International Society for Optics and Photonics, 2003.

[153] K. Iqbal, R. A. Salam, A. Osman, and A. Z. Talib, "Underwater image enhancement using an integrated colour model.," *IAENG International Journal of Computer Science*, vol. 34, no. 2, 2007.

[154] M. S. Hitam, E. A. Awalludin, W. N. J. H. W. Yussof, and Z. Bachok, "Mixture contrast limited adaptive histogram equalization for underwater image enhancement," in *2013 International conference on computer applications technology (ICCAT)*, pp. 1–5, IEEE, 2013.

[155] J. Y. Chiang and Y.-C. Chen, "Underwater image enhancement by wavelength compensation and dehazing," *IEEE transactions on image processing*, vol. 21, no. 4, pp. 1756–1769, 2011.

[156] H.-Y. Yang, P.-Y. Chen, C.-C. Huang, Y.-Z. Zhuang, and Y.-H. Shiau, "Low complexity underwater image enhancement based on dark channel prior," in *Second International Conference on Innovations in Bio-inspired Computing and Applications (IBICA)*, pp. 17–20, IEEE, 2011.

[157] C. Ancuti, C. O. Ancuti, C. De Vleeschouwer, R. Garcia, and A. C. Bovik, "Multi-scale underwater descattering," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 4202–4207, IEEE, 2016.

[158] Y.-T. Peng, X. Zhao, and P. C. Cosman, "Single underwater image enhancement using depth estimation based on blurriness," in *2015 IEEE International Conference on Image Processing (ICIP)*, pp. 4952–4956, IEEE, 2015.

[159] S. G. Narasimhan and S. K. Nayar, "Chromatic framework for vision in bad weather," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. 598–605, IEEE, 2000.

[160] Y. Y. Schechner, S. G. Narasimhan, and S. K. Nayar, "Instant dehazing of images using polarization," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 325, IEEE, 2001.

[161] S. G. Narasimhan and S. K. Nayar, "Contrast restoration of weather degraded images," *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 6, pp. 713–724, 2003.

[162] J. Kopf, B. Neubert, B. Chen, M. Cohen, D. Cohen-Or, O. Deussen, M. Uyttendaele, and D. Lischinski, "Deep photo: Model-based photograph enhancement and viewing," *ACM transactions on graphics (TOG)*, vol. 27, no. 5, pp. 1–10, 2008.

[163] T. Treibitz and Y. Y. Schechner, "Polarization: Beneficial for visibility enhancement?," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 525–532, IEEE, 2009.

[164] C. Ancuti, C. O. Ancuti, and C. De Vleeschouwer, "D-hazy: a dataset to evaluate quantitatively dehazing algorithms," in *IEEE International Conference on Image Processing (ICIP)*, pp. 2226–2230, IEEE, 2016.

[165] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang, "Single image dehazing via multi-scale convolutional neural networks," in *European conference on computer vision*, pp. 154–169, Springer, 2016.

[166] E. M. Alharbi, P. Ge, and H. Wang, "A research on single image dehazing algorithms based on dark channel prior," *Journal of Computer and Communications*, vol. 4, no. 02, p. 47, 2016.

[167] N. Carlevaris-Bianco, A. Mohan, and R. M. Eustice, "Initial results in underwater single image dehazing," in *Oceans 2010 Mts/IEEE Seattle*, pp. 1–8, IEEE, 2010.

[168] A. Galdran, D. Pardo, A. Picón, and A. Alvarez-Gila, "Automatic red-channel underwater image restoration," *Journal of Visual Communication and Image Representation*, vol. 26, pp. 132–145, 2015.

[169] H. Lu, Y. Li, L. Zhang, and S. Serikawa, "Contrast enhancement for images in turbid water," *JOSA A*, vol. 32, no. 5, pp. 886–893, 2015.

[170] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "Dehazenet: An end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.

[171] J. Zhang, J. Pan, W.-S. Lai, R. W. Lau, and M.-H. Yang, "Learning fully convolutional networks for iterative non-blind deconvolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3817–3825, 2017.

[172] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "Aod-net: All-in-one dehazing network," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4770–4778, 2017.

[173] W. Ren, L. Ma, J. Zhang, J. Pan, X. Cao, W. Liu, and M.-H. Yang, "Gated fusion network for single image dehazing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3253–3261, 2018.

[174] H. Zhang and V. M. Patel, "Density-aware single image de-raining using a multi-stream dense network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 695–704, 2018.

[175] C. Ancuti, C. O. Ancuti, C. De Vleeschouwer, and A. C. Bovik, "Night-time dehazing by fusion," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2256–2260, IEEE, 2016.

[176] L. Jiang, Y. Jing, S. Hu, B. Ge, and W. Xiao, "Deep refinement network for natural low-light image enhancement in symmetric pathways," *Symmetry*, vol. 10, p. 491, Oct 2018.

[177] C. Wei, W. Wang, W. Yang, and J. Liu, "Deep retinex decomposition for low-light enhancement," *arXiv preprint arXiv:1808.04560*, 2018.

[178] L. Shen, Z. Yue, F. Feng, Q. Chen, S. Liu, and J. Ma, "Msr-net: Low-light image enhancement using deep convolutional network," *arXiv preprint arXiv:1711.02488*, 2017.

[179] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang, "Enlightengan: Deep light enhancement without paired supervision," *arXiv preprint arXiv:1906.06972*, 2019.

[180] S. Lee, S. Yun, J.-H. Nam, C. S. Won, and S.-W. Jung, "A review on dark channel prior based image dehazing algorithms," *EURASIP Journal on Image and Video Processing*, vol. 2016, no. 1, p. 4, 2016.

[181] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, "Frequency-tuned salient region detection," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 1597–1604, IEEE, 2009.

[182] P. J. Burt and E. H. Adelson, "The laplacian pyramid as a compact image code," in *Readings in computer vision*, pp. 671–679, Elsevier, 1987.

[183] P. Burt and E. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on communications*, vol. 31, no. 4, pp. 532–540, 1983.

[184] T. Porto Marques, A. Rezvanifar, M. Cote, A. B. Albu, K. Ersahin, T. Mudge, and S. Gauthier, "Detecting marine species in echograms via traditional, hybrid, and deep learning frameworks," in *International Conference on Pattern Recognition (ICPR)*, 2020.

[185] D. G. Reid, "Report on echo trace classification," *ICES Cooperative Research Report*, no. 238, 2000.

[186] T. K. Stanton, "30 years of advances in active bioacoustics: A personal perspective," *Methods in Oceanography*, vol. 1, pp. 49–77, 2012.

[187] J. K. Horne, "Acoustic approaches to remote species identification: A review," *Fisheries Oceanography*, vol. 9, no. 4, pp. 356–71, 2000.

[188] D. Reid, C. Scalabrin, P. Petitgas, J. Masse, R. Aukland, P. Carrera, *et al.*, "Standard protocols for the analysis of school based data from echo sounder surveys," *Fisheries Research*, vol. 47, no. 2-3, pp. 125–36, 2000.

[189] P. LeFeuvre, G. Rose, R. Gosine, R. Hale, W. Pearson, and R. Khan, "Acoustic species identification in the Northwest Atlantic using digital image processing," *Fisheries Research*, vol. 47, no. 2-3, pp. 137–147, 2000.

[190] A. Charef, S. Ohshimo, I. Aoki, and N. Al Absi, "Classification of fish schools based on evaluation of acoustic descriptor characteristics," *Fisheries Science*, vol. 76, no. 1, pp. 1–11, 2010.

[191] S. Gauthier, J. Oeffner, and R. L. O'Driscoll, "Species composition and acoustic signatures of mesopelagic organisms in a subtropical convergence zone, the New Zealand Chatham Rise," *Marine Ecology Progress Series*, vol. 503, pp. 23–40, 2014.

[192] R. Proud, R. Mangeni-Sande, R. J. Kayanda, M. J. Cox, C. Nyamweya, C. Ongore, V. Natugonza, I. Everson, M. Elison, L. Hobbs, *et al.*, "Automated classification of schools of the silver cyprinid Rastrineobola argentea in Lake Victoria

acoustic survey data using random forests," *ICES Journal of Marine Science*, vol. 77, no. 4, pp. 1379–1390, 2020.

[193] L. Mannocci, Y. Baidai, F. Forget, M. T. Tolotti, L. Dagorn, and M. Capello, "Machine learning to detect bycatch risk: Novel application to echosounder buoys data in tuna purse seine fisheries," *Biological Conservation*, vol. 255, p. 109004, 2021.

[194] K. Malde, N. O. Handegard, L. Eikvil, and A.-B. Salberg, "Machine intelligence and the data-driven future of marine science," *ICES Journal of Marine Science*, p. fsz057, 2019.

[195] Y. Shang and J. Li, "Study on echo features and classification methods of fish species," in *10th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–6, IEEE, 2018.

[196] R. E. Schapire, "Explaining adaboost," in *Empirical inference*, pp. 37–52, Springer, 2013.

[197] H. Måløy, "Echobert: A transformer-based approach for behavior detection in echograms," *IEEE Access*, vol. 8, pp. 218372–218385, 2020.

[198] R. J. Korneliussen, Y. Heggelund, G. J. Macaulay, D. Patel, E. Johnsen, and I. K. Eliassen, "Acoustic identification of marine species using a feature library," *Methods in Oceanography*, vol. 17, pp. 187–205, 2016.

[199] A. Rezvanifar, T. P. Marques, M. Cote, A. B. Albu, A. Slonimer, T. Tolhurst, K. Ersahin, T. Mudge, and S. Gauthier, "A deep learning-based framework for the detection of schools of herring in echograms," in *NeurIPS Workshop Tackling Climate Change with Machine Learning*, 2019.

[200] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, 2017.

[201] G. French, M. Mackiewicz, M. Fisher, M. Challiss, P. Knight, B. Robinson, and A. Bloomfield, "Jellymonitor: automated detection of jellyfish in sonar images using neural networks," in *14th IEEE International Conference on Signal Processing (ICSP)*, pp. 406–412, IEEE, 2018.

[202] L. Liu, H. Lu, Z. Cao, and Y. Xiao, "Counting fish in sonar images," in *25th IEEE International Conference on Image Processing (ICIP)*, pp. 3189–3193, IEEE, 2018.

[203] D. Glukhov, R. Bohush, J. Mäkiö, and T. Hlukhava, "A joint application of fuzzy logic approximation and a deep learning neural network to build fish concentration maps based on sonar data," in *2nd International Workshop on Computer Modeling and Intelligent Systems (CMIS)*, pp. 133–142, 2019.

[204] D. Neupane and J. Seok, "A review on deep learning-based approaches for automatic sonar target recognition," *Electronics*, vol. 9, no. 11, 2020.

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

[206] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[207] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2117–2125, 2017.

[208] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 764–773, 2017.

[209] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1492–1500, 2017.

[210] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7263–7271, 2017.

[211] D. Mackas, "Seasonal cycle of zooplankton off southwestern british columbia: 1979–89," *Canadian Journal of Fisheries and Aquatic Sciences*, vol. 49, pp. 903––921, 1992.

[212] R. Tanasichuk, "Implications of interannual variability in euphausiid population biology for fish production along the southwest coast of vancouver island: a synthesis," *Fisheries Oceanography*, vol. 11, pp. 18–30, 2002.

[213] D. of Fisheries & Oceans, G. of Canada, F., and O. C., "Counting krill and other plankton." (accessed: 15.07.2021).

[214] D. Ware and G. McFarlane, "Climate-induced changes in pacific hake (merluccius productus) abundance and pelagic community interactions in the vancouver island upwelling system," *Canadian Special Publication of Fisheries and Aquatic Sciences*, vol. 121, pp. 509–521, 1995.

[215] T. Helser, I. Stewart, and O. Hamel, "Stock assessment of pacific hake (whiting) in u.s. and canadian waters in 2008," *Pacific Fishery Management Council*, p. 129, 2008.

[216] S. Romaine, D. Mackas, and M. Macaulay, "Comparison of euphausiid population size estimates obtained using replicated acoustic surveys of coastal inlets, and block average vs. geostatistical spatial interpolation methods," *Fisheries Oceanography*, vol. 11, no. 2, pp. 102–115, 2002.

[217] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, Springer, 2015.

[218] E. S. P. Ltd, "Home - echoview." https://echoview.com/. Accessed: 2021-08-16.

[219] A. Minelli, A. Tassetti, B. Hutton, G. Pezzuti Cozzolino, T. Jarvis, and G. Fabi, "Semi-automated data processing and semi-supervised machine learning for the detection and classification of water-column fish schools and gas seeps with a multibeam echosounder," *Sensors*, vol. 21, no. 9, p. 2999, 2021.

[220] C. Godínez-Pérez, H. Villalobos, D. Arizmendi-Rodríguez, V. González-Maynez, and A. Valdez-Pelayo, "Acoustic characterization of pacific hake (merluccius productus) aggregations from bi-frequency data in the gulf of california," in *IEEE/OES Acoustics in Underwater Geosciences Symposium (RIO Acoustics)*, pp. 1–4, IEEE, 2015.

[221] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[222] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning (ICML)*, pp. 448–456, PMLR, 2015.

[223] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[224] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[225] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[226] A. De Robertis and I. Higginbottom, "A post-processing technique to estimate the signal-to-noise ratio and remove echosounder background noise," *ICES Journal of Marine Science*, vol. 64, no. 6, pp. 1282–1291, 2007.

[227] M. Peña, "Incrementing data quality of multi-frequency echograms using the adaptive wiener filter (awf) denoising algorithm," *Deep Sea Research Part I: Oceanographic Research Papers*, vol. 116, pp. 14–21, 2016.

[228] R. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, 1973.

[229] M. Cote and A. Branzan Albu, "Texture sparseness for pixel classification of business document images," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 17, no. 3, pp. 257–273, 2014.

[230] M. Turner, "Texture discrimination by gabor functions," *Biological Cybernetics*, vol. 55, no. 2, pp. 71–82, 1986.

[231] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transac-*

tions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971–987, 2002.

[232] B. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[233] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and regression trees*. Routledge, 2017.

[234] C. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[235] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[236] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[237] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, no. 1, pp. 1–39, 2010.

[238] Z. Zhong, Z. Lin, X. Bidart, R.and Hu, I. Daya, Z. Li, W.-S. Zheng, J. Li, and A. Wong, "Squeeze-and-attention networks for semantic segmentation," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13065–13074, 2020.

[239] A. Kirillov, K. Wu, Y.and He, and R. Girshick, "Pointrend: Image segmentation as rendering," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9799–9808, 2020.

[240] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Learning a discriminative feature network for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1857–1866, 2018.

[241] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7151–7160, 2018.

[242] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2881–2890, 2017.

[243] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2015.

[244] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and A. Hartwig, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *European Conference on Computer Vision (ECCV)*, pp. 801–818, 2018.

[245] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1251–1258, IEEE, 2017.

[246] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[247] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*, pp. 740–755, Springer, 2014.

[248] "Pytorch." https://pytorch.org/. Accessed: 2021-12-21.

[249] "Torchvision 0.11.0 documentation." https://pytorch.org/vision/stable/models.html. Accessed: 2021-12-21.

[250] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[251] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[252] M. Everingham, S. Eslami, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes challenge: A retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, 2015.

[253] A. Abdulla, *Maritime traffic effects on biodiversity in the Mediterranean Sea. Volume 1: review of impacts, priority areas and mitigation measures*, vol. 1. IUCN, 2008.

[254] S. M. Nowacek, R. S. Wells, and A. R. Solow, "Short-term effects of boat traffic on bottlenose dolphins, tursiops truncatus, in sarasota bay, florida," *Marine Mammal Science*, vol. 17, no. 4, pp. 673–688, 2001.

[255] D. P. Nowacek, L. H. Thorne, D. W. Johnston, and P. L. Tyack, "Responses of cetaceans to anthropogenic noise," *Mammal Review*, vol. 37, no. 2, pp. 81–115, 2007.

[256] C. W. Clark, W. T. Ellison, B. L. Southall, L. Hatch, S. M. Van Parijs, A. Frankel, and D. Ponirakis, "Acoustic masking in marine ecosystems: intuitions, analysis, and implication," *Marine Ecology Progress Series*, vol. 395, pp. 201–222, 2009.

[257] M. M. Holt, D. P. Noren, V. Veirs, C. K. Emmons, and S. Veirs, "Speaking up: Killer whales (orcinus orca) increase their call amplitude in response to vessel noise," *The Journal of the Acoustical Society of America*, vol. 125, no. 1, pp. EL27–EL32, 2009.

[258] B. S. Halpern, M. Frazier, J. Afflerbach, J. S. Lowndes, F. Micheli, C. O'Hara, C. Scarborough, and K. A. Selkoe, "Recent pace of change in human impact on the world's ocean," *Scientific reports*, vol. 9, no. 1, pp. 1–8, 2019.

[259] R. Williams, A. J. Wright, E. Ashe, L. Blight, R. Bruintjes, R. Canessa, C. Clark, S. Cullis-Suzuki, D. Dakin, C. Erbe, *et al.*, "Impacts of anthropogenic noise on marine life: Publication patterns, new discoveries, and future directions in research and management," *Ocean & Coastal Management*, vol. 115, pp. 17–24, 2015.

[260] S. Bertazzon, P. D. O'Hara, O. Barrett, and N. Serra-Sogas, "Geospatial analysis of oil discharges observed by the national aerial surveillance program in the canadian pacific ocean," *Applied Geography*, vol. 52, pp. 78–89, 2014.

[261] L. M. Nichol, B. M. Wright, P. O'Hara, and J. K. Ford, "Risk of lethal vessel strikes to humpback and fin whales off the west coast of vancouver island, canada," *Endangered Species Research*, vol. 32, pp. 373–390, 2017.

[262] M. Robards, G. Silber, J. Adams, J. Arroyo, D. Lorenzini, K. Schwehr, and J. Amos, "Conservation science and policy applications of the marine vessel automatic identification system (ais)—a review," *Bulletin of Marine Science*, vol. 92, no. 1, pp. 75–103, 2016.

[263] L. Hermannsen, L. Mikkelsen, J. Tougaard, K. Beedholm, M. Johnson, and P. T. Madsen, "Recreational vessels without automatic identification system (ais) dominate anthropogenic noise contributions to a shallow water soundscape," *Scientific reports*, vol. 9, no. 1, pp. 1–10, 2019.

[264] G. of Canada, "SARA (Species at Risk Act): An act respecting the protection of wildlife species at risk in Canada," 2002. S.C. 2002, c. 29. SRKW listed as endangered in a 2010 amendment. Available at https://laws.justice.gc.ca/eng/acts/S-15.3/; accessed 25 September 2020.

[265] U. S. Fish and W. Service, "Endangered Species Act of 1973," 1973. SRKW listed as endangered in a 2005 amendment. Available at https://laws.justice.gc.ca/eng/acts/S-15.3/; accessed 25 September 2020.

[266] M. K. Pine, A. G. Jeffs, D. Wang, and C. A. Radford, "The potential for vessel noise to mask biologically important sounds within ecologically significant embayments," *Ocean & Coastal Management*, vol. 127, pp. 63–73, 2016.

[267] F. H. Jensen, L. Bejder, M. Wahlberg, N. A. Soto, M. Johnson, and P. T. Madsen, "Vessel noise effects on delphinid communication," *Marine Ecology Progress Series*, vol. 395, pp. 161–175, 2009.

[268] M. J. Williamson, A. S. Kavanagh, M. J. Noad, E. Kniest, and R. A. Dunlop, "The effect of close approaches for tagging activities by small research vessels on the behavior of humpback whales (megaptera novaeangliae)," *Marine Mammal Science*, vol. 32, no. 4, pp. 1234–1253, 2016.

[269] C. Smallwood, K. Pollock, B. Wise, N. Hall, and D. Gaughan, "Expanding roving-aerial surveys to include counts of recreational shore fishers from remotely-operated cameras: benefits, limitations and cost-effectiveness," *North American Journal of Fisheries Management*, vol. 32, pp. 1265–1276, 2012.

[270] B. W. Hartill, S. M. Taylor, K. Keller, and M. S. Weltersbach, "Digital camera monitoring of recreational fishing effort: Applications and challenges," *Fish and Fisheries*, vol. 21, no. 1, pp. 204–215, 2020.

[271] D. Lancaster, P. Dearden, D. R. Haggarty, J. P. Volpe, and N. C. Ban, "Effectiveness of shore-based remote camera monitoring for quantifying recreational fisher compliance in marine conservation areas," *Aquatic Conservation: Marine and freshwater ecosystems*, vol. 27, no. 4, pp. 804–813, 2017.

[272] U. Kanjir, H. Greidanus, and K. Oštir, "Vessel detection and classification from spaceborne optical images: A literature survey," *Remote sensing of environment*, vol. 207, pp. 1–26, 2018.

[273] T. Porto Marques and A. Branzan Albu, "L2uwe: A framework for the efficient enhancement of low-light underwater images using local contrast and multi-scale fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 538–539, 2020.

[274] C. Elvidge, M. Zhizhin, K. Baugh, and F.-C. Hsu, "Automatic boat identification system for viirs low light imaging data," *Remote Sensing*, vol. 7, no. 3, pp. 3020–3036, 2015.

[275] W. Krüger and Z. Orlov, "Robust layer-based boat detection and multi-target-tracking in maritime environments," in *2010 International WaterSide Security Conference*, pp. 1–7, IEEE, 2010.

[276] T.-H. Tran and T.-L. Le, "Vision based boat detection for maritime surveillance," in *2016 International Conference on Electronics, Information, and Communications (ICEIC)*, pp. 1–4, IEEE, 2016.

[277] X. Bao, S. Zinger, R. Wijnhoven, *et al.*, "Ship detection in port surveillance based on context and motion saliency analysis," in *Video Surveillance and Transportation Imaging Applications*, vol. 8663, p. 86630D, International Society for Optics and Photonics, 2013.

[278] J. Tang, C. Deng, G.-B. Huang, and B. Zhao, "Compressed-domain ship detection on spaceborne optical image using deep neural network and extreme learning machine," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 3, pp. 1174–1185, 2014.

[279] Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based cnn for ship detection," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 900–904, IEEE, 2017.

[280] R. Zhang, J. Yao, K. Zhang, C. Feng, and J. Zhang, "S-cnn-based ship detection from high-resolution remote sensing images.," *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, vol. 41, 2016.

[281] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[282] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162, 2018.

[283] D. D. Bloisi, A. Pennisi, and L. Iocchi, "Background modeling in the maritime domain," *Machine vision and applications*, vol. 25, no. 5, pp. 1257–1269, 2014.

[284] Z. Zou, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *arXiv preprint arXiv:1905.05055*, 2019.

[285] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1, pp. 886–893, IEEE, 2005.

[286] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, vol. 1, pp. I–I, IEEE, 2001.

[287] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[288] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE conference on computer vision and pattern recognition*, pp. 1–8, IEEE, 2008.

[289] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *2010 IEEE Computer society conference on computer vision and pattern recognition*, pp. 2241–2248, IEEE, 2010.

[290] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[291] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[292] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587, 2014.

[293] R. Girshick, "Fast r-cnn," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1440–1448, 2015.

[294] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[295] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37, Springer, 2016.

[296] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[297] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

[298] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, vol. 2, pp. 246–252, IEEE, 1999.

[299] A. Shimada, H. Nagahara, and R.-i. Taniguchi, "Background modeling based on bidirectional analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1979–1986, 2013.

[300] T. Minematsu, A. Shimada, and R.-i. Taniguchi, "Background initialization based on bidirectional analysis and consensus voting," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 126–131, IEEE, 2016.

[301] A. Kaehler and G. Bradski, *Learning OpenCV 3: computer vision in C++ with the OpenCV library*, pp. 397–404. " O'Reilly Media, Inc.", 2016.

[302] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.

[303] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1492–1500, 2017.

[304] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[305] P. C. Mahalanobis, "On the generalized distance in statistics," National Institute of Science of India, 1936.

[306] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 2, pp. 28–31, IEEE, 2004.

[307] Z. Zivkovic and F. Van Der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.

[308] G. of Canada Department of Fisheries and Oceans, "Review of the effectiveness of recovery measures for southern resident killer whales," November 2016. Online; accessed 25 September 2020.

[309] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[310] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, no. 4, pp. 13–es, 2006.

[311] R. Yu, H. Wang, and L. S. Davis, "Remotenet: Efficient relevant motion event detection for large-scale home surveillance videos," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1642–1651, IEEE, 2018.

[312] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[313] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

[314] A. El-Nouby, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, *et al.*, "Xcit: Cross-covariance image transformers," *arXiv preprint arXiv:2106.09681*, 2021.

[315] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," *arXiv preprint arXiv:2103.15808*, 2021.

[316] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.