Automatic Whale Detection using Deep learning

by

Muhammed Patel

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2024

© Muhammed Patel 2024

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

The following publications have resulted from the work presented in this thesis:

- Patel, M., Chen, X., Brubacher, N., Xu, L., & Clausi, D. A. (2022). Beluga whale detection from sliced aerial remote sensing images using object detection pipelines. Journal of Computational Vision and Imaging Systems, 8(1), 75-77.
- Patel, M., Chen, X., Xu, L., Cantu, F.J.P., Turnes, J.N., Brubacher, N.C., Clausi, D.A. and Scott, K.A., 2023, July. The Influence of Input Image Scale on Deep Learning-Based Beluga Whale Detection from Aerial Remote Sensing Imagery. In IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium (pp. 5732-5734). IEEE.

Abstract

Accurate monitoring of whale populations is essential for conservation efforts, yet traditional surveying methods are often time-consuming, expensive, and limited in coverage. This thesis investigates the automation of whale detection using state-of-the-art (SOTA) deep learning techniques applied to high-resolution aerial imagery. By leveraging advancements in computer vision, specifically object detection models, this research aims to develop a robust and efficient system for identifying and counting whales from aerial surveys. The study formulates whale detection as a small object detection problem and evaluates the performance of various SOTA models, including Faster R-CNN, YOLOv8, and Deformable DETR, paired with modern backbone architectures such as ConvNext-T, Swin-T, and ResNet-50. The influence of input image size and context on model performance is systematically explored by testing patch sizes ranging from 256 to 4096 pixels, marking this study as the first to investigate the efficacy of such large patch sizes in the remote sensing domain.

Results indicate that the Faster R-CNN model with a ConvNext-T backbone achieves the highest detection accuracy, with an average precision of 0.878 at an IoU threshold of 0.1, particularly when trained on larger patch sizes. The study also addresses the challenge of domain adaptation by implementing an active learning framework, designed to enhance model performance on new survey data with varying environmental conditions. A novel portfolio-based acquisition function, leveraging the social behavior of whales, is introduced to optimize the annotation process. This research significantly contributes to the field of automated whale monitoring, offering a scalable and adaptable solution that reduces annotation costs and improves the accuracy of population estimates. The developed system holds promise for enhancing conservation strategies and providing valuable insights into whale movements and behaviors.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Prof. Linlin Xu and Prof. David Clausi, for their unwavering support, invaluable guidance, and insightful feedback throughout the course of my research. Their expertise and encouragement have been instrumental in the successful completion of this thesis.

I am also profoundly grateful to the members of our research lab, who have contributed to this work in numerous ways. Xinwei, Neil, Fernando, and Javier, your camaraderie, intellectual discussions, and continuous support have made this journey both enjoyable and enriching. Your contributions and encouragement have been pivotal to my research endeavors.

Lastly, I would like to extend my appreciation to everyone who has supported me in any capacity during this project. Thank you all for being a part of this journey.

Dedication

To my loving family.

Table of Contents

Author's Declaration ii

Statement of Contributions iii

Abstract iv

Acknowledgements v

Dedication vi

List of Figures ix

List of Tables xii

List of Abbreviations xiii

1 Introduction 1

1.1 Background 2

1.2 Problem Statement 4

1.3 Research questions 4

1.4 Significance of the study 4

1.5 Related Works 5

1.6 Structure of the thesis 8

2 Deep Learning Based Whale Detection **9**
2.1 Dataset . . . 9
2.2 Methodology . . . 13
2.2.1 Experimental Design . . . 18
2.3 Implementation . . . 19
2.4 Results . . . 22
2.4.1 Effect of Models and Backbones . . . 22
2.4.2 Effect of patch size . . . 25
2.4.3 Results of best model . . . 29

3 Adapting to New Domain: Case Study **32**
3.0.1 Designing acquisition function . . . 34
3.0.2 Implementation: Active Learning Workflow . . . 37
3.0.3 Result . . . 39

4 Conclusion **44**
4.1 Conclusion . . . 44
4.2 Limitations and future work . . . 46

References **48**

List of Figures

1.1 Humpback whale entangled in pot gear near Western Australia. Image credit: Department of Environment and Conservation, Australia. 2

1.2 An example of aerial imagery taken from a helicopter. 3

2.1 Survey locations in Canadian Arctic 11

2.2 Appearance of beluga whales in different backgrounds. Best viewed when zoomed in. Zoom in here. 13

2.3 Appearance of narwhals in different backgrounds. Best viewed when zoomed in. Zoom in here. 14

2.4 (a) DFO annotation in purple points, (b) bounding box conversion. 14

2.5 Schematics of dataset preparation showing the number of images in train, test, and validation set. The no-whale images are randomly sampled uniformly from all the no-whale images. 15

2.6 Modern object detection model components. The modules in the clouds are generally interchangeable. 17

2.7 Patches from size 256 to 4096. Orientation of the waves is critical to discern whales from waves. Best viewed when zoomed in. Zoom in here. 19

2.8 Comparison of Effective Receptive Fields (ERF)[45] between different backbone models. Pixels with higher intensity indicate larger responses regarding the central pixel. Figure adapted from [40], used under CC BY 4.0. 20

2.9 Schematic showing the logic to determine TP, FP for GT and prediction boxes using IoU threshold. Box 1 and Box 2 are prediction boxes. 21

2.10 Effect of models on patch size 256. P-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9. 24

2.11 Effect of models on patch size 512 and 1024. P-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9. 25

2.12 Effect of models on patch size 2048 and 4096. The p-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9. . . 26

2.13 Effect of patch size on FasterRCNN with (a) ConvNext-T, (b) Swin-T. The P-R curve is plotted on the left side. The right side represents a spider plot showing R2 and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9. 27

2.14 Prediction of FasterRCNN with ConvNext-T backbone on various patch sizes. Increasing the patch size helps the model eliminate false positives due to waves. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here. 28

2.15 Prediction of FasterRCNN with ConvNext-T backbone on various backgrounds. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here. 29

2.16 Predictions on images overlooked by DFO but found by our best model. Figure (a) contains around 2000 whales, and (b) contains around 865 whales. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here. 31

3.1 Map showing the survey transects over the Canadian High Arctic region. . 32

3.2 Icy conditions in the 2023 survey not present in earlier surveys. 33

3.3 The small ice shards in 2023 surveys closely resemble the beluga whales of the 2014 survey. Best viewed when zoomed in. Zoom in here. 34

3.4 Bowhead whale which is not present in the earlier surveys missed by our model. Best viewed when zoomed in. Zoom in here. 35

3.5 Beluga whales detected with low confidence along with some clear whales not predicted, using our best model without any fine-tuning. Best viewed when zoomed in. Zoom in here. 36

3.6 Overview of active learning paradigm. 37

3.7 Summary of differences in appearance between images from old and new survey. 38

3.8 Cummulative distribution of whales per image from old surveys sorted in descending order. 20% of images owns 80% of total whales. 39

3.9 Portfolio-based acquisition function 40

3.10 Active Learning Workflow implemented using Label Studio 41

3.11 The labelstudio annotation interface. 42

3.12 Image with fewer than five objects acquired using our acquisition function. (a) shows the model's predictions before fine-tuning. (b) shows the predictions after fine-tuning. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in.Zoom in here. 42

3.13 Image with more than 20 objects acquired using our acquisition function. (a) shows the model's predictions before fine-tuning. (b) shows the predictions after fine-tuning. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here. 43

4.1 A whale is detected near the path of a ship 45

4.2 Seals detected by our best model without any finetuning on the new 2023 survey. 47

List of Tables

2.1 The survey year, location, and an example of whales from each survey. The purple bounding box indicates the location of whales in challenging examples. 10

2.2 Survey statistics indicating the number of images, images with whales, and the total number of individual whales found. 12

2.3 Breakdown by dataset of number of whales in training (Train), testing (Test), and Validation (Val) datasets. 16

2.4 Model hyperparameters 22

2.5 $AP_{IoU=10\%}$ of all the models and backbones across various patch sizes presented in this study. The top five models are highlighted in bold. 23

3.1 Performance metrics across four active learning iterations, showing the Average Precision at IoU=10 ($AP_{IoU=10}$), True Positives (TP), False Positives (FP), and False Negatives (FN). The metrics are reported on the test set images after fine-tuning on the corresponding training images. 40

List of Abbreviations

AP average precision 20, 22

CNN convolutional neural networks 3, 6, 14, 15

DFO Fisheries and Oceans Canada 4, 5, 9, 12, 19, 32

FN false negative 20, 21

FP false positive 20

FPN feature pyramid network 15, 18

GT ground truth 20

IoU intersection over Union 20

mAP mean average precision 20

TP true positive 20, 21

Chapter 1

Introduction

Whales, as some of the largest and most charismatic marine mammals, play a crucial role in ocean ecosystems and have significant cultural and economic value worldwide. However, accurate monitoring of whale populations remains a challenge due to their vast oceanic habitats and often remote locations. Traditional whale surveying methods, such as visual surveys from ships or planes, are time-consuming, expensive, and limited in their coverage. In recent years, the advent of high-resolution aerial and satellite imagery has opened new possibilities for whale monitoring at larger scales.

This thesis explores the automation of whale detection using SOTA object detection models on aerial images. By leveraging advances in computer vision and deep learning, the goal is to develop a robust and efficient system for identifying and counting whales from aerial surveys. This system has the potential to significantly enhance our ability to monitor whale populations, track their movements, and inform conservation efforts.

The primary objectives of this research are:

- formulate whale detection as an object detection problem;
- develop and study the effects of different SOTA models on whale detection;
- study the influence of input of image size and context on model's performance; and
- extend the developed pipeline for large-scale whale monitoring on new surveys via active learning.

1.1 Background

Whale Monitoring and Conservation

Whale populations worldwide have faced numerous threats, including historical commercial whaling, ship strikes[7], entanglement in fishing gear[25], and climate change impacts on their habitats. Accurate population estimates and distribution data are crucial for effective conservation strategies. Traditional survey methods, though valuable, are constrained in their spatial and temporal coverage due to their manual nature[4], emphasizing the need for more efficient and extensive automatic monitoring techniques.



Figure 1.1: Humpback whale entangled in pot gear near Western Australia. Image credit: Department of Environment and Conservation, Australia.

Remote Sensing for Whale Detection

The use of aerial and satellite imagery for whale detection has gained traction in recent years. Very high-resolution (VHR) satellite imagery, with resolutions of 50 cm or less, has made it possible to identify individual whales from space[18]. This non-invasive approach offers several advantages, including the ability to survey large areas, access remote locations, and reduce disturbance to whale populations compared to ship or manual surveys. Fig. 1.2 shows an example of an image captured at an altitude of 2000 ft.

Object Detection in Computer Vision

Object detection is a computer vision task that involves identifying and locating specific objects within an image or video. In recent years, deep learning-based approaches, like

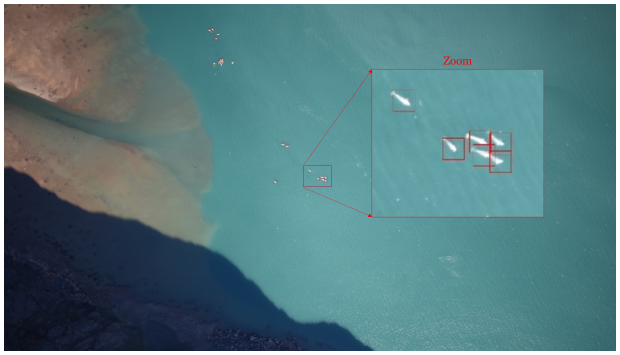


Figure 1.2: An example of aerial imagery taken from a helicopter.

convolutional neural networks (CNN) have achieved SOTA performance in object detection tasks[35]. Several popular object detection architectures have been developed, including: 1) R-CNN (region-based CNN) and its variants (Fast R-CNN[15], Faster R-CNN[51]); 2) YOLO (You Only Look Once)[22]; 3) SSD (Single Shot Detector)[39]; 4) DeTr (Detection transformer)[6]. These models have shown remarkable success in various domains, including autonomous driving, facial recognition, and medical imaging [35].

Application to Whale Detection

Applying object detection models to whale detection in aerial imagery presents unique challenges:

1. Variability in whale appearance: Whales can appear differently based on species, behavior (e.g., surfacing, diving), and environmental conditions.
2. Aerial images are often captured with cameras lacking radiometric calibration, which can result in variations due to factors such as water turbidity, sun glint, and atmospheric conditions.
3. Scale: Whales may occupy only a tiny portion of high-altitude images.
4. Limited training data: Compared to other object detection tasks, there are fewer annotated datasets of whales in aerial imagery.

Despite these challenges, recent studies have demonstrated the potential of deep learning-based approaches for whale detection. For example, Lee et al. [30] used a CNN model-based classification to detect the beluga whales.

This thesis builds upon these foundations, aiming to develop and evaluate an object detection model specifically tailored for whale detection in aerial imagery. By addressing the unique challenges of this domain and leveraging SOTA object detection architectures, we seek to contribute to the growing field of automated whale monitoring from remote sensing data.

1.2 Problem Statement

The objective is to develop an automated whale detection model for the Fisheries and Oceans Canada (DFO) to facilitate aerial counting. The model must achieve high precision due to the nature of aerial surveys, where only a few images contain whales among a large dataset. After consultation with the DFO, the model must have a minimum 90% detection rate (recall). Additionally, the model must be adaptable to different domains, as each survey occurs under varying environmental conditions. The developed pipeline should be easily extendable to other species and datasets, ensuring versatility and scalability for broader applications.

1.3 Research questions

1. How do SOTA object detection models perform on aerial survey data collected by the DFO?
2. Given the large size of aerial images, what is the ideal patch size required for optimal performance of the detection model?
3. How does the receptive field of the model influence its performance on aerial survey images?
4. What strategies can be employed to adapt the model to new domains with varying environmental conditions?

1.4 Significance of the study

The automated whale detection model will significantly reduce annotation costs and improve the accuracy of abundance estimates. Even in a semi-automated setting, the model

will assist the DFO by filtering out irrelevant images, thereby streamlining the analysis process. This is particularly beneficial given the often non-explainable nature of deep learning models, where human oversight can enhance reliability and trust in the results.

This study also addresses a gap in the literature concerning the influence of patch size on object detection in large aerial images. By being the first to explore this aspect, this contributes valuable insights to the field.

Additionally, the development of a model that can automatically adapt to new domains through active learning ensures that the solution remains effective across various environmental conditions and datasets, further enhancing its utility and robustness.

1.5 Related Works

Whale detection has long been an important area of research for marine biologists and conservationists. Various techniques have been developed over the years to locate and monitor whale populations. These range from visual surveys conducted by human observers[56] to more advanced technological approaches using acoustic monitoring[46] and satellite imagery[20, 30]. The choice of detection method often depends on factors like the species being studied, environmental conditions, and available resources.

Methods of Whale Detection

Historically, whale detection relied heavily on visual observation methods[47]. These included ship-based surveys, where researchers would conduct transects from boats, visually scanning the ocean surface for whale sightings; aerial surveys, in which planes or helicopters would fly predetermined routes to spot whales from above; and land-based observations, where fixed stations on coastlines would track whale movements, especially during migration seasons. While these methods provided valuable data, they were limited by factors such as weather conditions, observer fatigue, and the inability to detect submerged whales[3, 20]. Acoustic monitoring using hydrophones emerged as an important complementary technique, allowing detection of whale vocalizations underwater[46]. However, acoustic methods also have limitations in terms of range and species specificity[12].

Recently, aerial and satellite image surveys have emerged as powerful tools for whale detection, offering broader coverage, increased efficiency, and reduced disturbance to whale populations. High-resolution satellite imagery and aerial photographs enable researchers to survey vast ocean areas rapidly, including remote or dangerous regions inaccessible to

traditional methods. These techniques provide valuable data on whale behavior, distribution patterns, and habitat use without the disruption caused by close-range observations. However, these methods face challenges such as limited ability to detect submerged animals, potential weather-related obstacles, and the labor-intensive process of annotating images covering large geographical areas. Despite these limitations, aerial and satellite surveys represent a significant advancement in whale research and conservation efforts, complementing traditional detection methods.

Introduction to Deep Learning

Deep learning is a subset of machine learning that uses artificial neural networks with multiple layers to extract higher-level features from raw input progressively. These deep neural networks can automatically learn complex patterns in data, making them particularly well-suited for tasks like image recognition and classification[29]. The rapid advancement of deep learning techniques, coupled with increases in computational power and availability of large datasets, has led to breakthroughs in many fields, including computer vision and natural language processing[53].

Deep Learning Applications in Wildlife Detection

The application of deep learning to wildlife detection and monitoring has gained significant traction in recent years. Researchers have successfully employed CNN and other deep learning architectures to detect and classify various animal species in images and videos[58]. These applications span a wide range of scenarios, from automatic identification of individual animals in camera trap images[59] to detection and counting of birds in aerial surveys[8, 1], and tracking animal movements in video footage[23]. In the marine domain, deep learning has shown promise in detecting dolphins in underwater video and classifying whale calls in acoustic recordings[57, 44, 14]. These successes have paved the way for advanced whale detection methods. For instance, Guirado et al.[20] developed a two-stage network for whale detection, where the first stage filters whale-containing patches, and the second stage localizes them. Similar approaches have been applied to other marine species, such as Dujon et al.'s[13] work on counting sea turtles. The application of deep learning extends beyond marine environments. Delplanque et al.[10] developed a point-based CNN to detect and count animals in African landscapes, while Gonçalves et al.[16] used an ensemble of semantic segmentation models to detect seals from Worldview 3 satellite images. These diverse applications demonstrate the versatility and potential of deep learning in wildlife monitoring across different ecosystems. To address concerns about the reliability

of AI-based detection systems, some researchers have adopted a human-in-the-loop approach. Boulent et al.[4] and Kulits et al.[27] implemented systems where all predictions are manually reviewed. This method is gaining traction among biologists, who appreciate its efficiency in promptly correcting misclassifications while maintaining human oversight. The success of deep learning in these related areas underscores its potential to revolutionize whale detection methods, offering improved accuracy, efficiency, and scalability in marine wildlife monitoring.

Gaps in the Literature

While deep learning has shown promise in wildlife detection, including marine species, several critical gaps persist in the current literature.

1. Biased Evaluation Metrics: Aerial surveys often produce a vast quantity of images, with only a small percentage—typically around 5%—containing the targeted objects of interest. Many studies report their precision and recall solely on this small subset of target-containing images or some even on small sliced patches. This approach can be misleading as it does not reflect the model's performance on the entire dataset, which predominantly consists of non-target images. Real-world surveys often contain lookalikes such as sun glare, ice, swell, rocks, and vegetation. These elements can be misclassified as the target species. Even a small false positive rate can lead to a significant number of misclassifications, potentially overwhelming human reviewers and reducing the efficiency of the detection process. We aim to address this gap by reporting our metrics on full images, not sliced. This approach is highly significant because, unlike other studies, we provide a comprehensive assessment of our model's performance on the entire aerial image dataset, including the vast majority of images without targets. Due to the computational demands of evaluating datasets in the terabyte range, we first identify our best model and then report the full dataset-level precision only for this model. Meanwhile, for all the other models, we still report the performance on a randomly selected subset of negative images.

2. Limitations of Image Slicing: Due to memory constraints, slicing has become the standard preprocessing technique for training deep-learning models on large aerial imagery. Prior research predominantly relies on 256 $\times$ 256 pixel slices or up to 1024 $\times$ 1024 in newer studies for model training. This approach leads to problems such as significant imbalance between positive and negative patches and may cause the model to lose the necessary context. With recent improvements in computational hardware

(with GPU memory now up to 80 GB), it is possible to train models on much larger image inputs, potentially enhancing performance as suggested by Zhu et. al[61], Patel et. al [50] and Li et. al[32]. Our study explores the efficacy of slicing aerial images for object detection tasks by testing various patch sizes, ranging from 256×256 to 4096×4096 pixels, and utilizing full images where possible. We also investigate the influence of patch size on various SOTA models and backbones, aiming to understand how much context is needed to reduce false positives in large aerial images with many lookalikes while maintaining high recall. To the best of our knowledge, this study represents the first comprehensive investigation into the influence of patch size on object detection performance in the context of aerial imagery analysis.

By addressing these limitations, we aim to contribute to more robust and reliable deep-learning methods for wildlife detection in aerial imagery. Our findings will provide researchers with valuable insights to select appropriate models and backbones tailored to their specific needs, balancing factors such as computational resources, image size, and detection accuracy.

1.6 Structure of the thesis

Chapter 2 introduces the dataset used for whale detection, presents various object detection architectures, explores different backbone networks, and details experiments evaluating the impact of different patch sizes on detection performance. This chapter also discusses the rationale behind the model selections and provides a comprehensive analysis of the results obtained.

Chapter 3 presents a case study on adapting the developed whale detection model to a 2023 survey using active learning techniques. This chapter details the design of a custom acquisition function, outlines the software implementation for deploying the active learning workflow, and analyzes the results of the fine-tuned model. It demonstrates the model's adaptability to new datasets and highlights the effectiveness of the active learning approach.

Chapter 2

Deep Learning Based Whale Detection

This chapter introduces the dataset used, model design, object detection models, various backbones, and the experiments assessing different patch sizes.

2.1 Dataset

Our research utilizes four surveys conducted by the DFO across the Canadian Arctic (see Fig. 2.1) between 2014 and 2017. The surveys were performed using an aircraft equipped with a 36.15 MP Nikon D810 camera, fitted with a 25 mm lens, to capture aerial images. These images were geocoded using an onboard GPS unit at a target altitude of 610 meters, with a capture interval of approximately 4 seconds. Each image has a resolution of 4912 $\times$ 7360 pixels, with a spatial resolution of approximately 10 cm/pixel.

Table 2.1 summarizes each survey's locations and species examples, while Fig. 2.1 illustrates the approximate locations where the photographs were taken.

Dataset Exploration

Table 2.2 summarizes the data from whale surveys conducted in 2014, 2015, 2016, and 2017, detailing the size of data collected, the number of images, whale findings, and images containing whales. The total size of all surveys is 1019.7 GB, comprising 39,492 images,

Table 2.1: The survey year, location, and an example of whales from each survey. The purple bounding box indicates the location of whales in challenging examples.

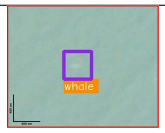
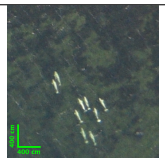
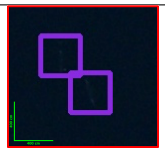
Year	Location	Species	Example Image
2014	Cumberland Sound	Beluga	
2015	West Hudson Bay	Beluga	
2016	Eclipse Sound	Narwhal	
2017	Cumberland Sound	Beluga	



Figure 2.1: Survey locations in Canadian Arctic

58,292 whale findings, and 1,644 images containing whales. Notably, the 2015 WHB survey contains the highest number of whale findings, with a total of 47,459. The number of images containing whales is very sparse, leading to a significant class imbalance between positive and negative images.

The background appearance in the images varies significantly across the dataset due to environmental factors affecting the water's appearance. These factors include the time of day, wind speed, weather conditions, swell size, sediment, viewing angle, and water turbidity. These variations affect the visibility and appearance of the belugas and narwhals. Fig. 2.2 shows the appearance of beluga whales under different environmental conditions. The sun glint in Fig. 2.2(a), closely resembles the appearance of whales. Similarly, the appearance of narwhals is depicted in Fig. 2.3, with the white spots in Fig. 2.3 (b) resembling beluga whales.

The ideal model should be capable of accounting for these highly non-linear variations. The non-linear modeling capabilities of deep learning make it particularly suitable for this

Table 2.2: Survey statistics indicating the number of images, images with whales, and the total number of individual whales found.

Survey	Species	# Images	# Images with Whales	# Whale findings
2014 CS Beluga	Beluga	18,718	467	1935
2015 WHB Beluga	Beluga	5,440	278	47459
2017 CS Beluga	Beluga	5,232	233	3717
2016 ES Narwhal	Narwhal	10,102	666	5181
Total		**39,492**	**1644**	**58,292**

dataset. Consequently, addressing these challenges and developing a robust whale detection model becomes a complex task.

Dataset preprocessing

The annotation of each image was manually performed by experts at DFO using point-based annotations, mainly into four categories: Adult, Calf, Juvenile, Unknown, and Questionable. For simplicity, we map the categories to a single label (i.e., 'whales'), in all our experiments. Next, the annotation points were converted into bounding boxes of size 40×40 pixels, or 400 cm×400 cm, with the whale at the center. This size was chosen based on the average length of belugas and narwhals, which is approximately 4 meters[49]. The bounding box visualization of whales is shown in Fig. 2.4. Formulating the whale detection problem as an object detection task rather than point detection facilitates the application of a wide range of well-established object detection models without requiring architectural adaptations.

Out of the 39,492 total images across all surveys, only 1,644 images contain whales. To address the significant class imbalance and reduce the dataset size, we first collected all the positive images from each year's survey. We then added negative images, randomly sampled at twice the number of positive images. Finally, the dataset was split into training, testing, and validation sets with a ratio of 65:20:15. The schematic of the dataset preparation is shown in Fig. 2.5, and the number of whales in each group is shown in Table 2.3. We will use these images in all our experiments.

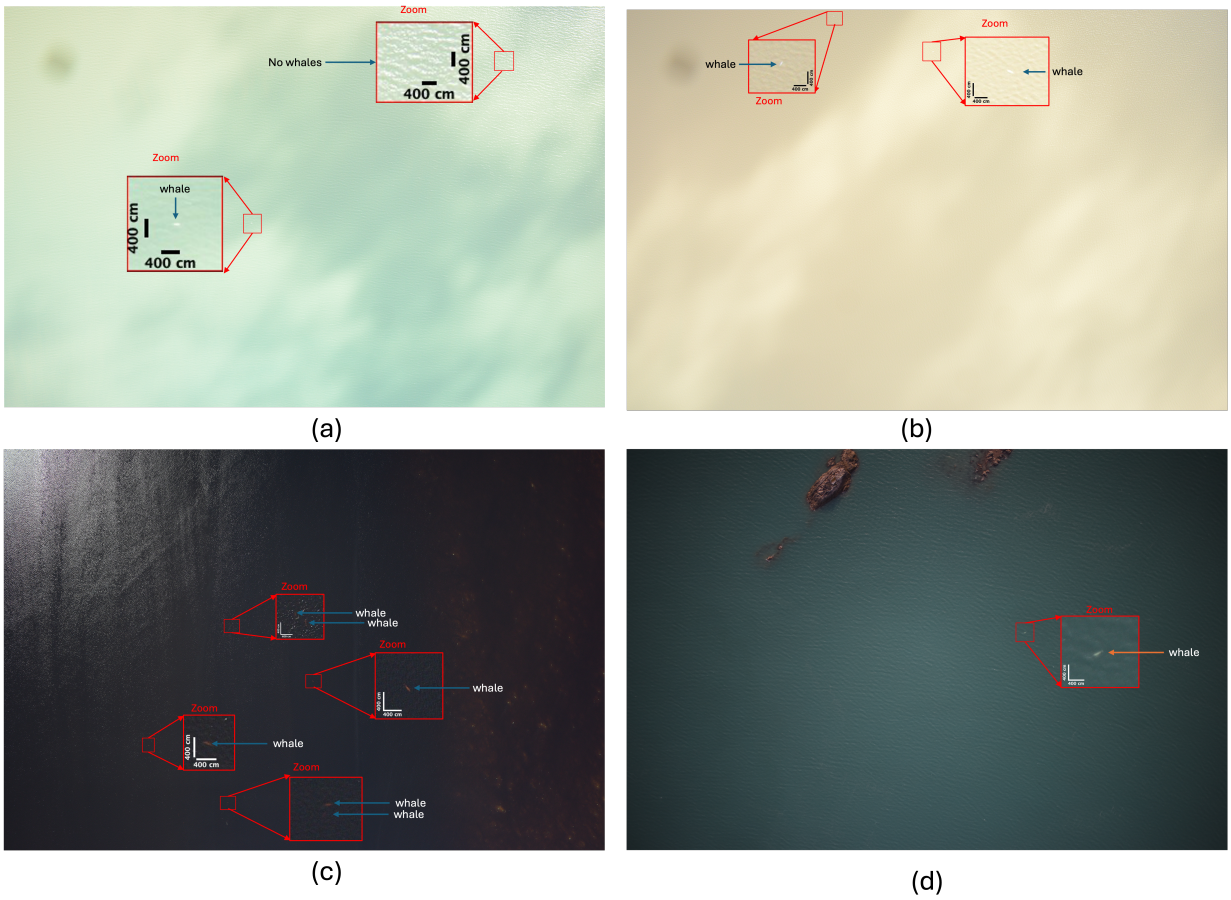


Figure 2.2: Appearance of beluga whales in different backgrounds. Best viewed when zoomed in. Zoom in here.

2.2 Methodology

This study employs a systematic experimental approach to evaluate the efficacy of various factors influencing the performance of whale detection models. Our investigation focuses on three primary variables: patch size, backbone architectures, and object detection models. Before delving into the experimental methodology, it is essential to elucidate some fundamental concepts of object detection models.



Figure 2.3: Appearance of narwhals in different backgrounds. Best viewed when zoomed in. Zoom in here.

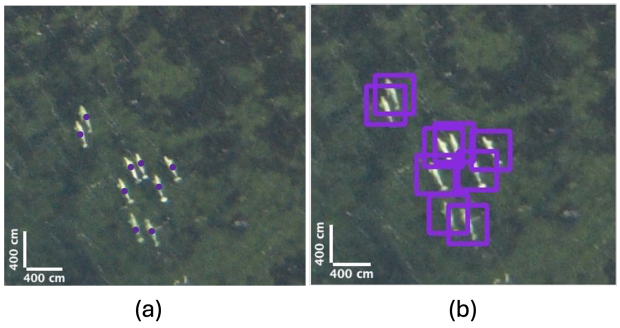


Figure 2.4: (a) DFO annotation in purple points, (b) bounding box conversion.

Object Detection Model

Object detection is a computer vision technique that identifies and locates objects within images or videos. In recent years, deep learning-based methodologies, particularly those utilizing CNNs and transformers, have surpassed traditional computer vision approaches in both accuracy and computational efficiency. These advanced models have revolutionized object detection, achieving SOTA performance across various benchmark datasets. Contemporary object detection architectures typically comprise three principal components: the backbone, neck, and head. The model components are illustrated in Fig. 2.6.

Backbone

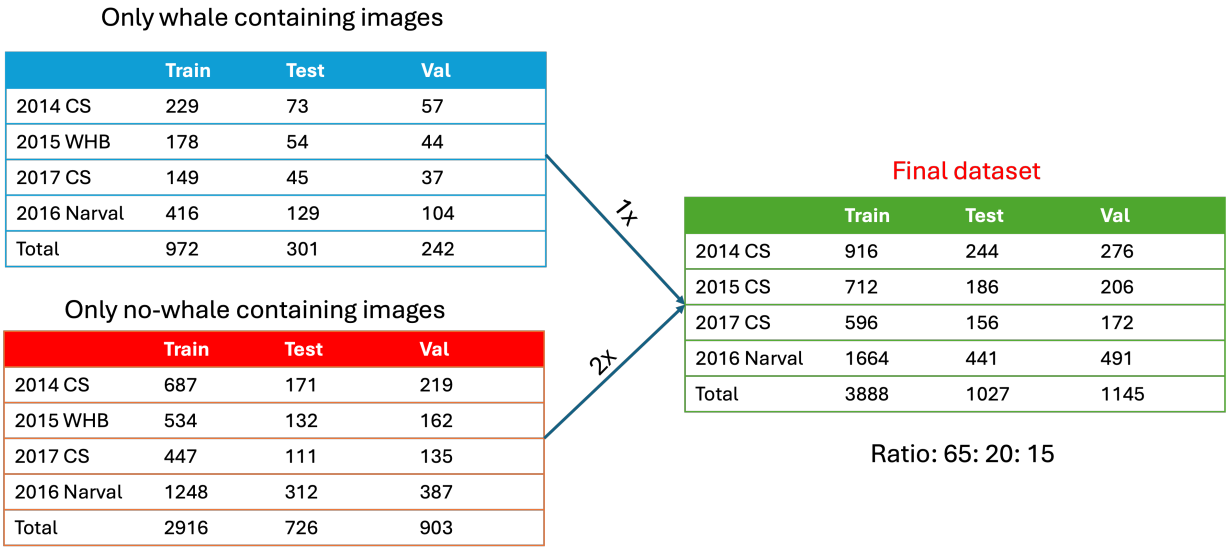


Figure 2.5: Schematics of dataset preparation showing the number of images in train, test, and validation set. The no-whale images are randomly sampled uniformly from all the no-whale images.

The backbone serves as the feature extraction network within the object detector, responsible for deriving high-level features from the input image [51]. Commonly employed backbones include pre-trained CNNs such as ResNet [21], VGG [55], or more recent architectures such as Swin Transformer [41] and ConvNeXt [42]. The selection of an appropriate backbone significantly influences both the detector's performance and its computational efficiency.

Neck The neck component is designed to process and refine the features extracted by the backbone. It typically incorporates a series of layers that enhance feature representation, facilitating the aggregation of features at multiple scales. This multi-scale approach improves the detection of objects across various sizes. Prominent neck structures include feature pyramid network (FPN)[33] and Path Aggregation Networks (PANet)[38].

Head

The head component performs the final detection task, processing the refined features from the neck to generate predictions[9]. These predictions typically include bounding box coordinates, object class, and confidence levels. The architecture of the head varies among different object detectors. For instance, Faster R-CNN employs separate heads for region proposal and classification, while YOLO utilizes a single head for both tasks simultaneously,

Table 2.3: Breakdown by dataset of number of whales in training (Train), testing (Test), and Validation (Val) datasets.

	Train (# of whales)	Test (# of whales)	Val (# of whales)
2014 CS Beluga	1009	507	159
2015 WHB	27763	8000	10746
2016 Narwhal	2409	608	586
2017 CS Beluga	3454	948	727
Total	34635	10063	12218

exemplifying a key distinction between two-stage and single-stage detectors.

The experimental framework of this study incorporates three families of object detectors, each representing a distinct paradigm in object detection:

Two-Stage Detector: We employ Faster R-CNN [51], a widely adopted method renowned for its accuracy in object detection tasks. Its two-stage approach, first proposing regions of interest and then classifying these regions, renders it particularly effective for detecting whales in large aerial images.

One-Stage Detector: We use YOLOv8[24] for this category. YOLO (You Only Look Once) models are known for their speed and efficiency, performing detection in a single stage, which is beneficial for real-time applications.

Transformer-Based Detector: DETR (Detection Transformer) [6] is a Transformer-based object detection model. DETR has gained attention for its end-to-end object detection capability without the need for handcrafted components, leveraging the power of transformers for spatial understanding. We use Deformable DETR[62], a variant of DETR designed to handle complex object detection tasks. Deformable DETR enhances the original transformer-based approach with deformable attention mechanisms, allowing for better handling of objects at various scales and improving detection accuracy in large and complex images.

The backbone component also plays a particularly significant role in determining the model's performance. The next section introduces the backbone architectures used in this study.

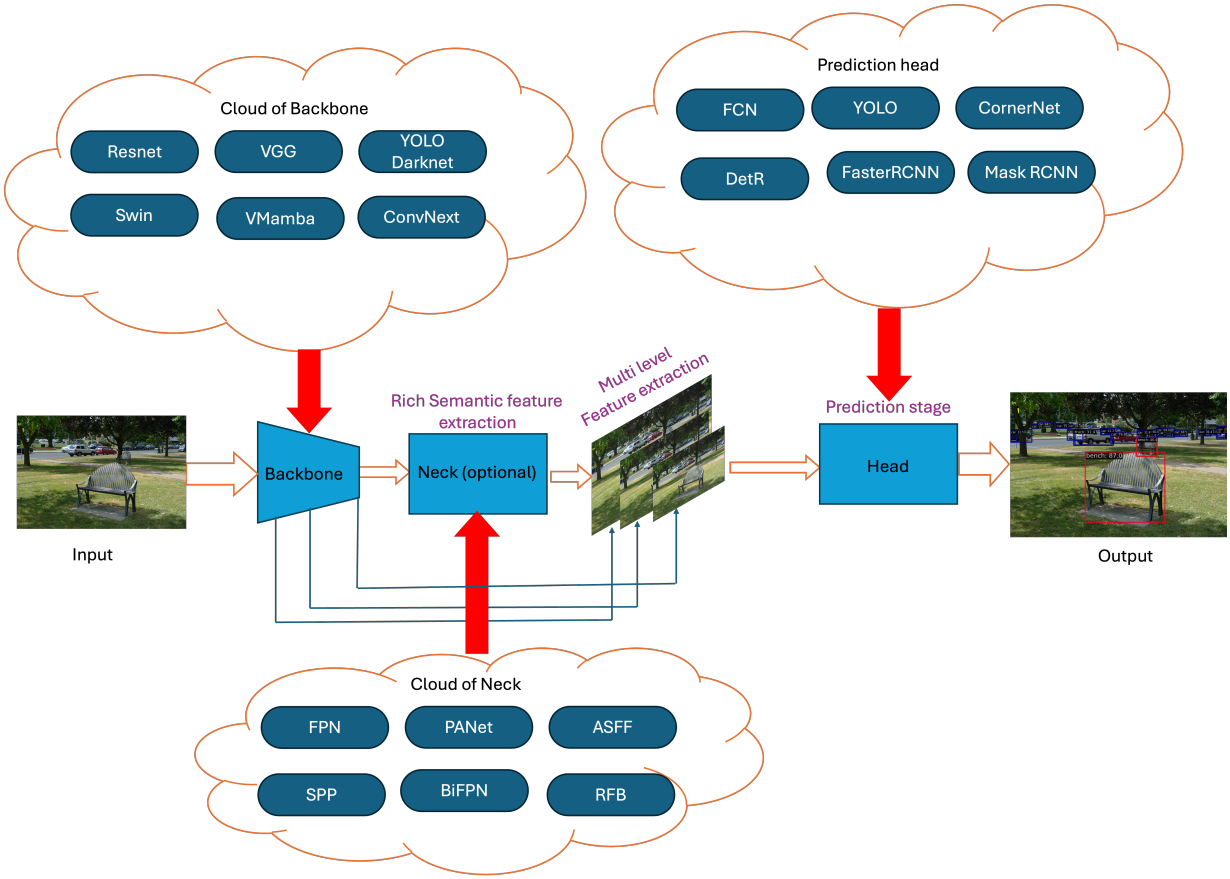


Figure 2.6: Modern object detection model components. The modules in the clouds are generally interchangeable.

Backbone Architectures

To comprehensively evaluate the impact of feature extraction on whale detection, we compare the performance of several well-established backbone architectures namely:

ResNet-50[21]: A staple in computer vision tasks, ResNet-50 is included due to its widespread use and proven effectiveness in various image classification and detection tasks.

Swin-T[41]: The Swin Transformer, introduced in 2021, was developed to address limitations of traditional vision transformers in handling high-resolution images and to improve computational efficiency while maintaining high performance in vision tasks. It introduces a hierarchical structure with linear computational complexity concerning image size by

computing local windowed attention, making it suitable for handling large aerial images efficiently.

ConvNeXT-T[42]: Inspired by transformer design principles, ConvNeXt modernizes the original ResNet architecture through transformer-inspired modifications. Its inclusion enables us to assess the benefits of such modernized convolutional networks in the context of whale detection.

VMamba[40]: VMamba, introduced in 2024, extends the Mamba state-space language model[19] to computer vision, achieving linear computational complexity while maintaining a global receptive field. It employs a 2D Selective Scan (SS2D) module to bridge the gap between 1D selective scan and 2D vision data, enabling efficient contextual information gathering from various perspectives. Because of its global context-capturing capabilities, it is a valuable candidate for our study.

2.2.1 Experimental Design

We would like to systematically evaluate the impact of three key factors on automatic whale detection: patch size, object detection models, and backbone architectures. We train and test each combination of these factors to comprehensively assess their individual and combined effects on detection performance.

We investigate five patch sizes: 256, 512, 1024, 2048, and 4096 pixels. This range allows us to determine the optimal contextual information for accurate whale detection. We expect that larger patch sizes may provide more contextual cues, potentially improving detection precision, especially for situations where there are whale look-alike. However, excessively large patches might introduce unnecessary background noise and increase computational load, potentially reducing overall recall. Fig. 2.7 illustrates the significance of contextual information in an image affected by sunglare, revealing that the orientation—a crucial factor for distinguishing whales from waves—becomes discernible at a patch size of 512 pixels, albeit clarity is not fully achieved.

Different object detection models are tested to evaluate their efficacy in handling the unique challenges of whale detection, such as varying scales and understanding context. We anticipate that models with multi-scale feature representations, such as Faster R-CNN with FPN[33], may perform better due to their ability to detect objects at different scales.

Including various backbone architectures allows us to assess the impact of different feature extraction capabilities on whale detection. Deeper and more complex backbones might extract more nuanced features, potentially improving detection accuracy, but at the

cost of increased computational memory requirements. We expect that backbones with higher receptive field may perform better. Fig. 2.8 shows the effective receptive field [45] of the backbones used in this study.

By systematically testing these factors, we aim to: a) Determine the optimal patch size that balances contextual information and computational efficiency. b) Assess how different object detection models handle the complexities of whale detection in aerial imagery. c) Evaluate the effectiveness of various backbone architectures in extracting relevant features for whale detection. d) Identify the optimal configuration for operational deployment in marine mammal monitoring programs conducted by the DFO.



Figure 2.7: Patches from size 256 to 4096. Orientation of the waves is critical to discern whales from waves. Best viewed when zoomed in. Zoom in here.

2.3 Implementation

To implement our experiments, we utilized the mmdetection framework[9], a comprehensive toolbox for object detection. All experiments were conducted using two NVIDIA A100 (40GB) Tensor Core GPUs on a high-performance computing (HPC) cluster provided by the Digital Research Alliance of Canada.

We pre-generate only the training image patches from the original images. Whereas for inference, the patches are sliced on the fly. The predictions for these individual patches are subsequently integrated using a non-max suppression[48] to produce coherent detection results for the entire image. To address class imbalance in our dataset, we employed an undersampling technique during the training phase. This approach ensured that the model

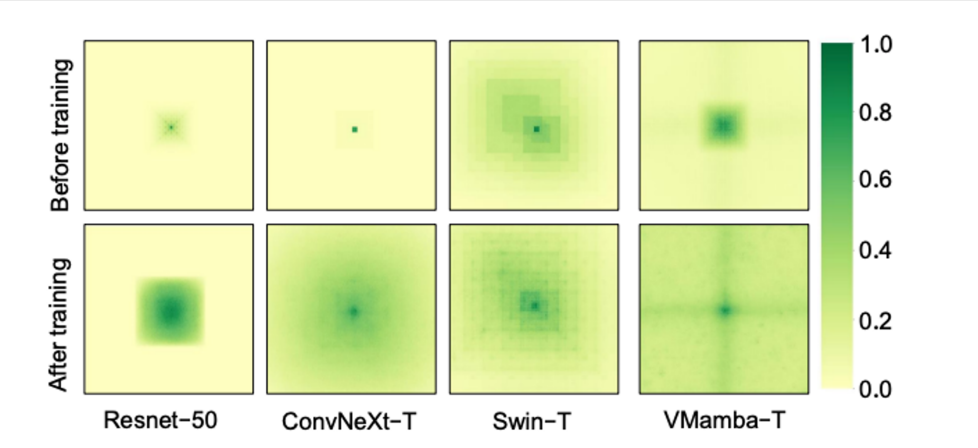


Figure 2.8: Comparison of Effective Receptive Fields (ERF)[45] between different backbone models. Pixels with higher intensity indicate larger responses regarding the central pixel. Figure adapted from [40], used under CC BY 4.0.

sees positive (whale-containing) and negative (whale-free) patches equally, mitigating potential bias in the model's learning process.

Our data augmentation strategy is implemented using the Albumentations library[5]. These augmentations include a variety of image-space transformations such as RandomBrightnessContrast, HueSaturationValue shift, JpegCompression, and ChannelShuffle. These image transformations are particularly useful in enabling the model to detect whales in diverse backgrounds. Additionally, we applied simple geometric transformations, including flipping and rotation. These augmentations enhance the model's generalization capabilities and robustness to various image conditions.

The optimal learning rates for our models were determined through a grid search, and the specific hyperparameters are presented in Table 2.4. All backbone weights are initialized from ImageNet [11], while the model weights are initialized with pre-trained weights from the MSCOCO dataset[34].

For evaluation, we use the MSCOCO-based [34] metrics to assess object detection performance. These metrics primarily comprise mean average precision (mAP), precision, and recall, which are calculated by computing intersection over Union (IoU) between ground truth (GT) and predicted bounding boxes. Fig. 2.9 illustrates the process of calculating true positive (TP) and false positive (FP) for a single image. This methodology is then applied across the entire dataset to compute aggregate TP, FP, and false negative (FN), from which precision and recall are subsequently calculated. average precision (AP) is calculated

as the area under the precision-recall curve generated by varying the model's confidence threshold. If there are multiple classes, a mean AP across different classes is computed. However, this conventional approach has a limitation: it classifies predictions as FN even when the predicted and ground truth boxes are in close proximity. In human-in-the-loop methodology, where each prediction undergoes manual verification, it is probable that an expert observer would identify such near-miss predictions. To address this limitation, we introduce two custom metrics:

(a) A metric based on the Euclidean distance between centroids of predicted and ground truth bounding boxes, accounting for near-miss detections. This metric classifies predictions as TP if they fall within a specified Euclidean distance threshold (e.g., 1000 pixels) from the ground truth centroid. The resulting precision and recall using this criterion are denoted as P2 and R2, respectively.
(b) A quantification of images containing only FP detections without any ground truth annotations. This metric highlights the potential additional workload for human annotators in reviewing extraneous images.

These custom metrics complement the standard COCO metric, providing a more comprehensive assessment in the context of whale detection.

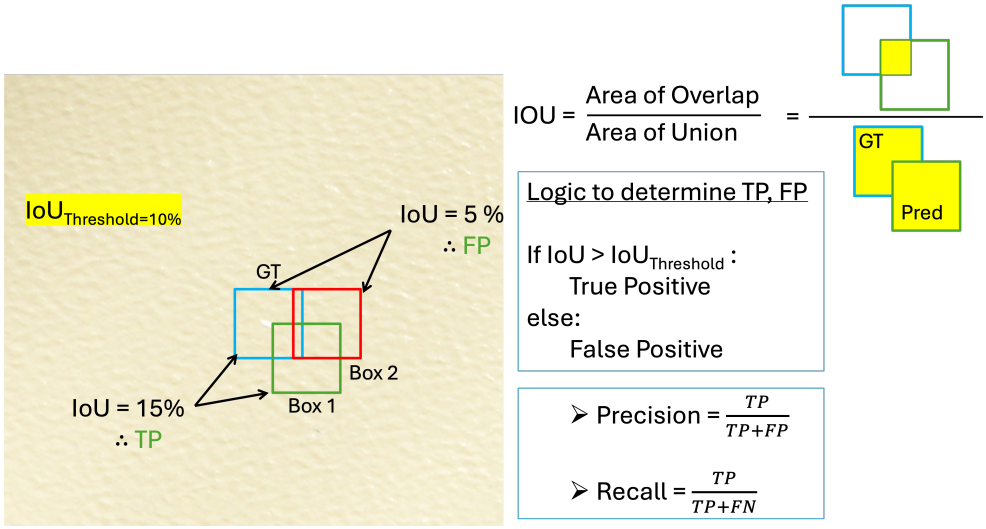


Figure 2.9: Schematic showing the logic to determine TP, FP for GT and prediction boxes using IoU threshold. Box 1 and Box 2 are prediction boxes.

Table 2.4: Model hyperparameters

Optimizer	SGD or AdamW[43] (Depends on the model/backbones used)
Batch size	Depends on the patch size to ensure full GPU utilization
Number of iterations	40,000
lr schedule	MultiStepLR

2.4 Results

To compare the performance of various patch sizes, we plot the conventional precision-recall curve and calculate the AP at a fixed Intersection over a Union (IoU) threshold of 0.1. This IoU threshold is chosen with the understanding that a biologist will manually verify each prediction, and a model's performance is considered satisfactory as long as it roughly locates the object. Table 2.5 shows the $AP_{IoU=10}$ of all the experiments conducted in this study. The FasterRCNN model with ConvNext-T backbone generally performs better and has the best model with patch size 4096, achieving an $AP_{IoU=10}$ of 0.878. For FasterRCNN, the performance tends to improve with the increase of patch size, whereas, for Deformable DETR and Yolov8, the performance gains are reduced beyond patch size 1024. Several models, particularly with larger patch sizes (4096), encounter OOM errors, indicating GPU memory limitations when dealing with large patches.

In the subsequent subsections, we present a comparative analysis of different backbones and models across various patch sizes. We also present additional metrics, including P2 and R2 at a Euclidean threshold of 1000 pixels, as well as the number of images without ground truth but with predictions and the number of false positives in such images.

2.4.1 Effect of Models and Backbones

In this section, we present a comparative analysis of various models and backbones across different patch sizes. Figures 2.10 and 2.12 illustrate the precision-recall curves and spider plots, providing a comprehensive view of the performance metrics. Our investigation reveals distinct patterns in the performance of different backbone architectures:

1. Swin-T and ConvNext-T: These modern architectures consistently demonstrate superior performance across the spectrum of patch sizes. When integrated with Faster-RCNN, they achieve the highest $AP_{IoU=10}$ scores, notably:

Table 2.5: $AP_{IoU=10\%}$ of all the models and backbones across various patch sizes presented in this study. The top five models are highlighted in bold.

# Params	Model	Backbone	Patch size 256	Patch size 512	Patch size 1024	Patch size 2048	Patch size 4096	resize 50%
40.10M	Deformable DETR	ResNet-50	0.794	0.820	0.847	0.761	NP	0.318
38.84M	Deformable DETR	Swin-T	0.00	0.841	0.828	NP	NP	NP
40.80M	Deformable DETR	VMamba-T	0.775	0.788	LDC	NP	NP	NP
39.15M	Deformable DETR	ConvNext-T	0.812	0.833	0.832	0.741	NP	NP
41.35M	FasterRCNN	ResNet-50	0.746	0.81	0.833	0.843	0.84	0.663
44.75M	FasterRCNN	Swin-T	0.793	0.841	**0.863**	**0.86**	NP	NP
46.71M	FasterRCNN	VMamba-T	0.772	0.788	0.818	0.802	NP	NP
45.05M	FasterRCNN	ConvNext	0.784	0.834	**0.861**	**0.866**	**0.878**	NP
11.14M	Yolov8-s	Yolov8CSPDarkNet	0.747	0.791	0.812	0.645	NP	NP
43.69M	Yolov8-l	Yolov8CSPDarkNet	0.746	0.804	0.802	0.731	NP	NP

Note: NP (Not possible) represents that it couldn't be completed because of OOM (out of memory) error. LDC represents loss did not converge (even after hyperparameter tuning).

- At 1024 patch size: Swin-T (0.863) and ConvNext-T (0.861)
- At 2048 patch size: ConvNext-T (0.866) and Swin-T (0.860). ConvNext-T exhibits remarkable scalability, maintaining high performance even at 4096 patch size (0.878 with FasterRCNN), while Swin-T encounters computational limitations at this scale.

2. ResNet-50: This established architecture exhibits consistent performance across various patch sizes and model integrations. It demonstrates particular efficacy with Deformable DETR at 1024 patch size (0.847) and with FasterRCNN at larger scales (0.843 at 2048, 0.840 at 4096). While it may not consistently achieve the peak performance of Swin-T or ConvNext-T, its stability across scales is noteworthy.

3. VMamba-T: This backbone generally underperforms relative to Swin-T, ConvNext-T, and ResNet-50 across all patch sizes despite its claim of having a global receptive field. Notably, it failed to converge with Deformable DETR at a 1024 patch size, indicating potential instability or optimization challenges.

4. Yolov8CSPDarkNet: Utilized exclusively with Yolov8 models, this backbone demonstrates moderate performance. It achieves optimal results at 1024 patch size (0.812 for Yolov8-s, 0.802 for Yolov8-l) but exhibits a significant performance degradation at larger patch sizes, particularly at 2048 (0.645 for Yolov8-s, 0.731 for Yolov8-l).

For models, Deformable DETR is generally the second-best performer, nearly trailing behind Faster R-CNN(from patch 256 to 1024). The gap between Deformable DETR and

Faster R-CNN gets larger when the patch size exceeds 1024. This may be attributed to the presence of the region proposal network (RPN) and the utilization of anchor boxes in Faster R-CNN, which ensures that the network scans all image regions equally. In contrast, Deformable DETR relies on a query-based detection framework, and the learned queries may not be very effective in detecting small objects across large images. Another possible reason is the size of the feature maps passed to the encoder, which is too low, and due to attention scaling quadratically, increasing the size of the feature maps may not be possible with current hardware.

Yolov8-s and Yolov8-l also show similar behavior as Deformable DETR, with performance increasing until a patch size of 1024 and then starting to decline. The reason could be that YOLO is also anchor-free, meaning objects are detected directly from the feature maps. We suspect the lack of anchors might lead to insufficient localization precision for larger patch sizes. As a result, the model produces too many false positives(see spider plots in Fig. 2.11 and 2.12), often confusing whales with similar-looking objects.

In conclusion, this analysis demonstrates that Faster R-CNN, especially when paired with modern backbone architectures, provides the best performance and scalability across various patch sizes. This architecture is particularly well-suited for small object detection tasks.

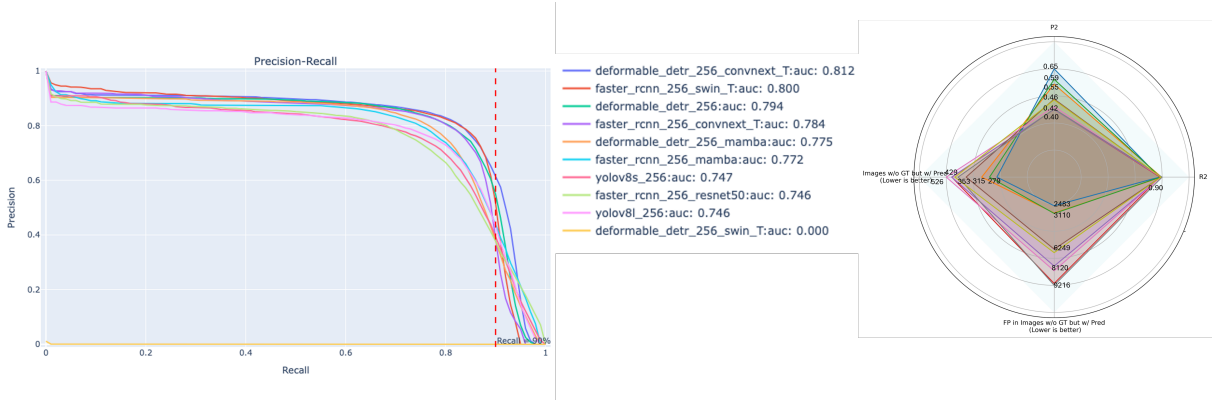


Figure 2.10: Effect of models on patch size 256. P-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9.

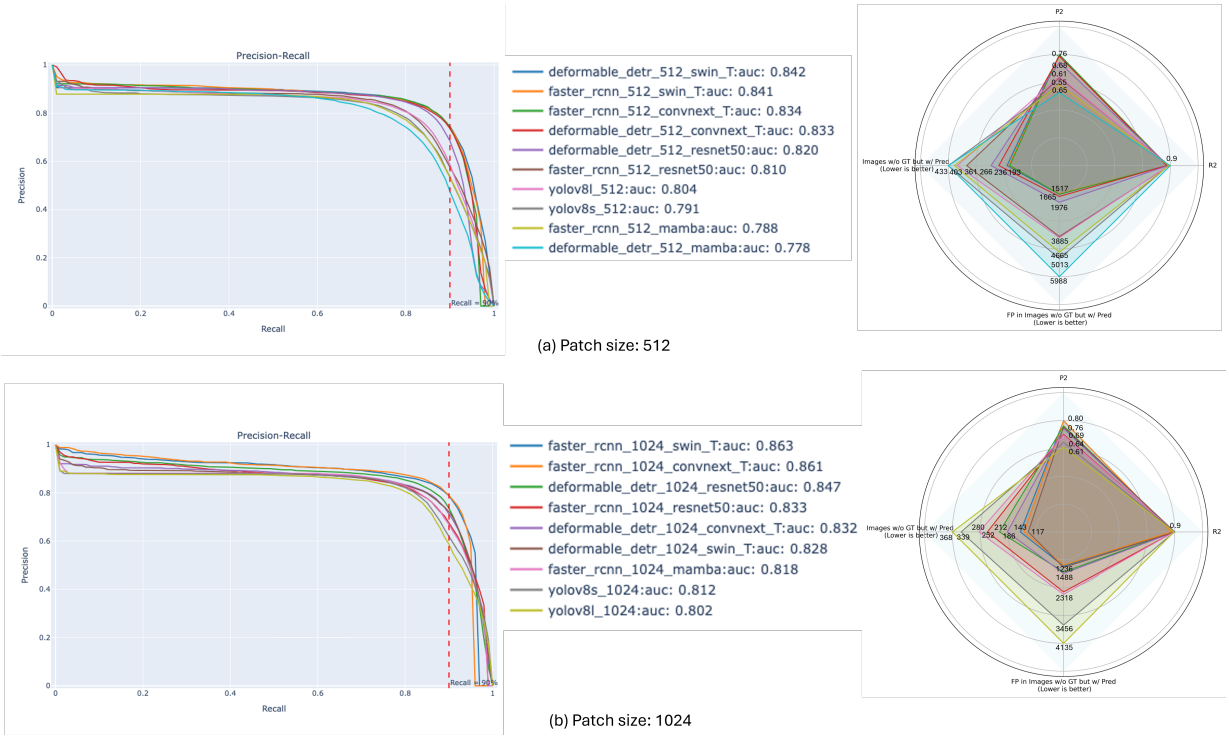


Figure 2.11: Effect of models on patch size 512 and 1024. P-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9.

2.4.2 Effect of patch size

In this section, we compare the effect of increasing the patch size on the best model architecture so far, i.e FasterRCNN with ConvNext-T as shown in Fig. 2.13, increasing the patch size decreases the amount of false positives and the best model achevies a remarkable AP of 0.878 with only 83 images containing extra predictions out of 726 non-whale images. Fig. 2.14 shows a prediction visualization for various patch size of FasterRCNN with ConvNext-T. The zoom-in patch is the size of the patch used by the model to make prediction, observe how increasing the patch size helps model increase the context and helps reducing the false positives.

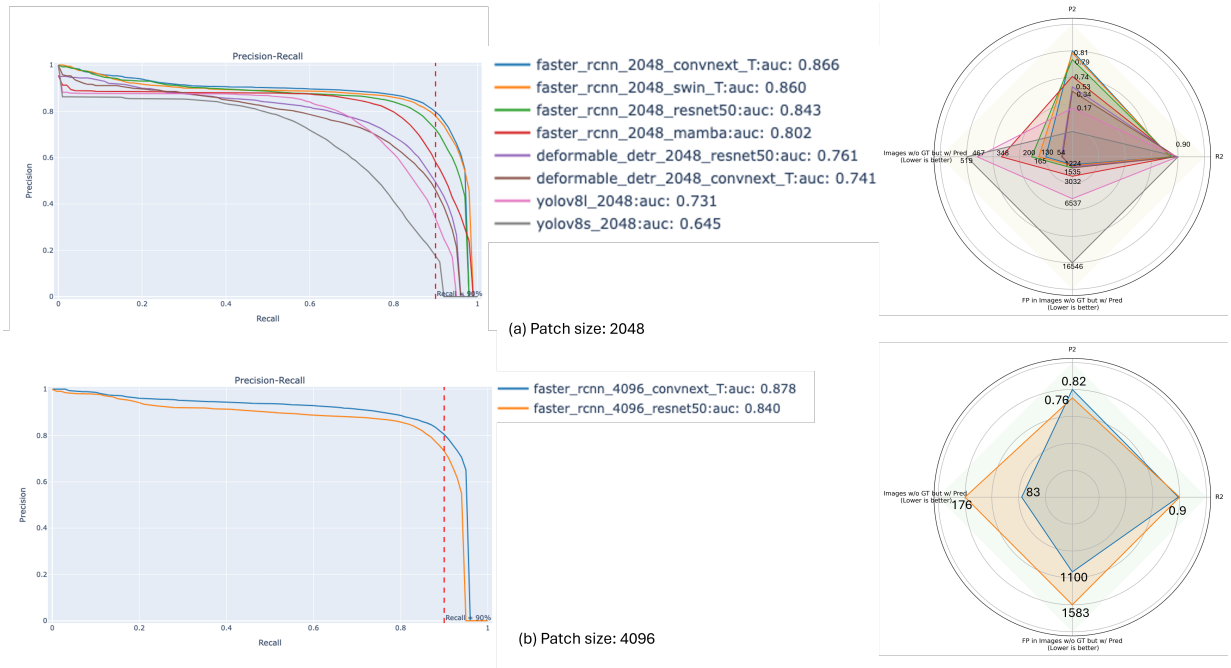


Figure 2.12: Effect of models on patch size 2048 and 4096. The p-R curve is plotted on the left side. The right side represents a spider plot showing R2, and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9.

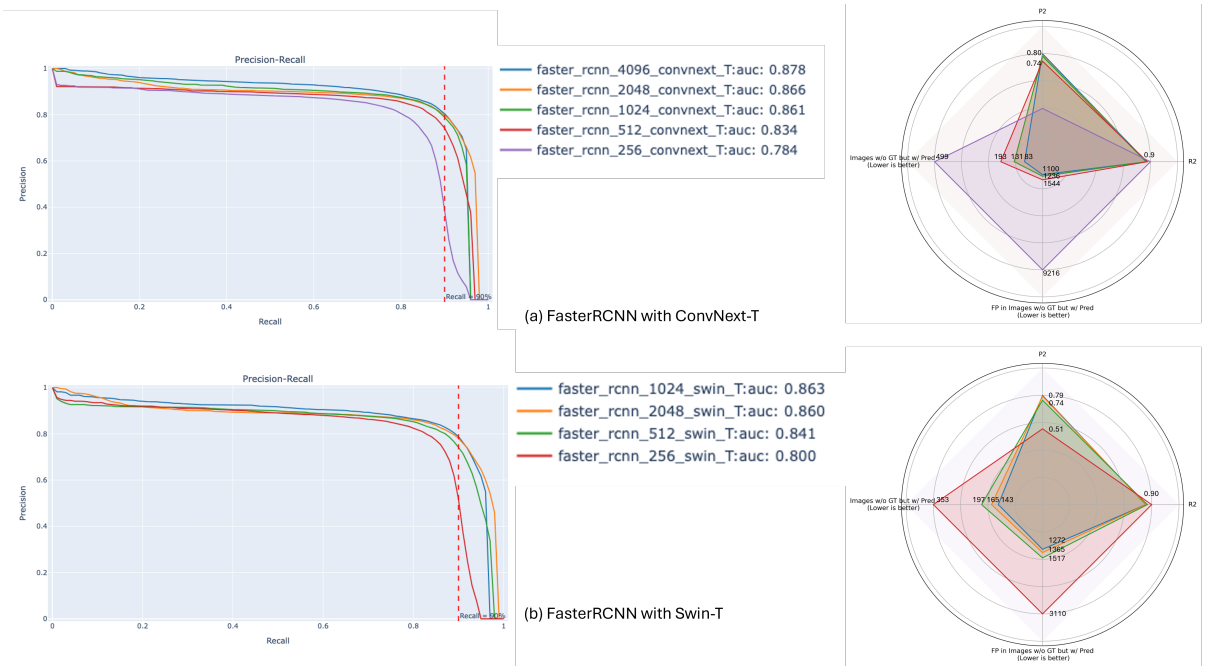


Figure 2.13: Effect of patch size on FasterRCNN with (a) ConvNext-T, (b) Swin-T. The P-R curve is plotted on the left side. The right side represents a spider plot showing R2 and P2, images with predictions but without ground truth, and the number of false predictions in such images. The dashed red line represents recall = 0.9.

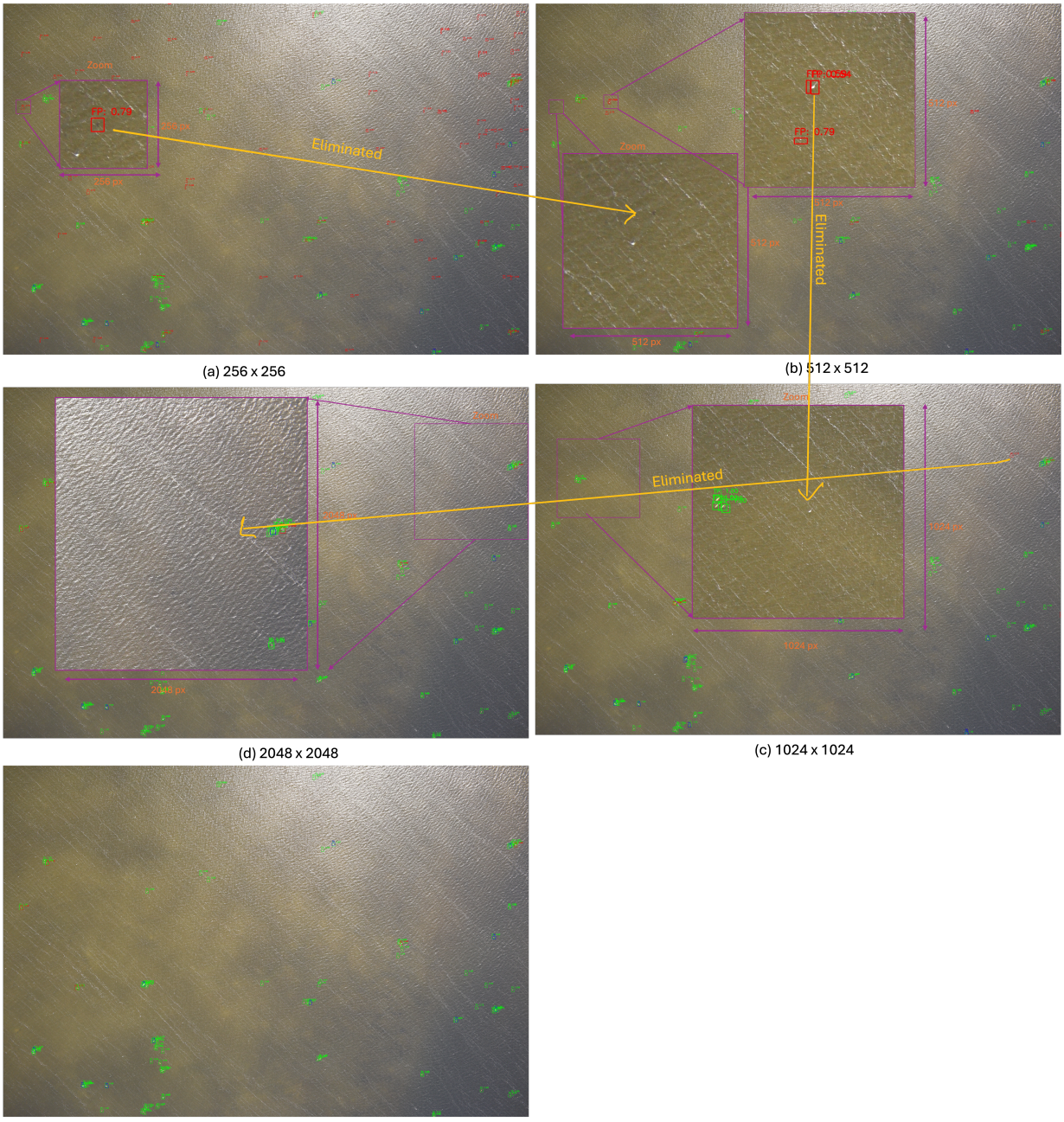


Figure 2.14: Prediction of FasterRCNN with ConvNext-T backbone on various patch sizes. Increasing the patch size helps the model eliminate false positives due to waves. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here.

2.4.3 Results of best model

The optimal model configuration was determined to be FasterRCNN with a ConvNext backbone, trained on 4096 patch sizes, achieving an $AP_{IoU=10}$ of 0.878. Fig. 2.15 illustrates the model's ability to detect whales in diverse backgrounds.

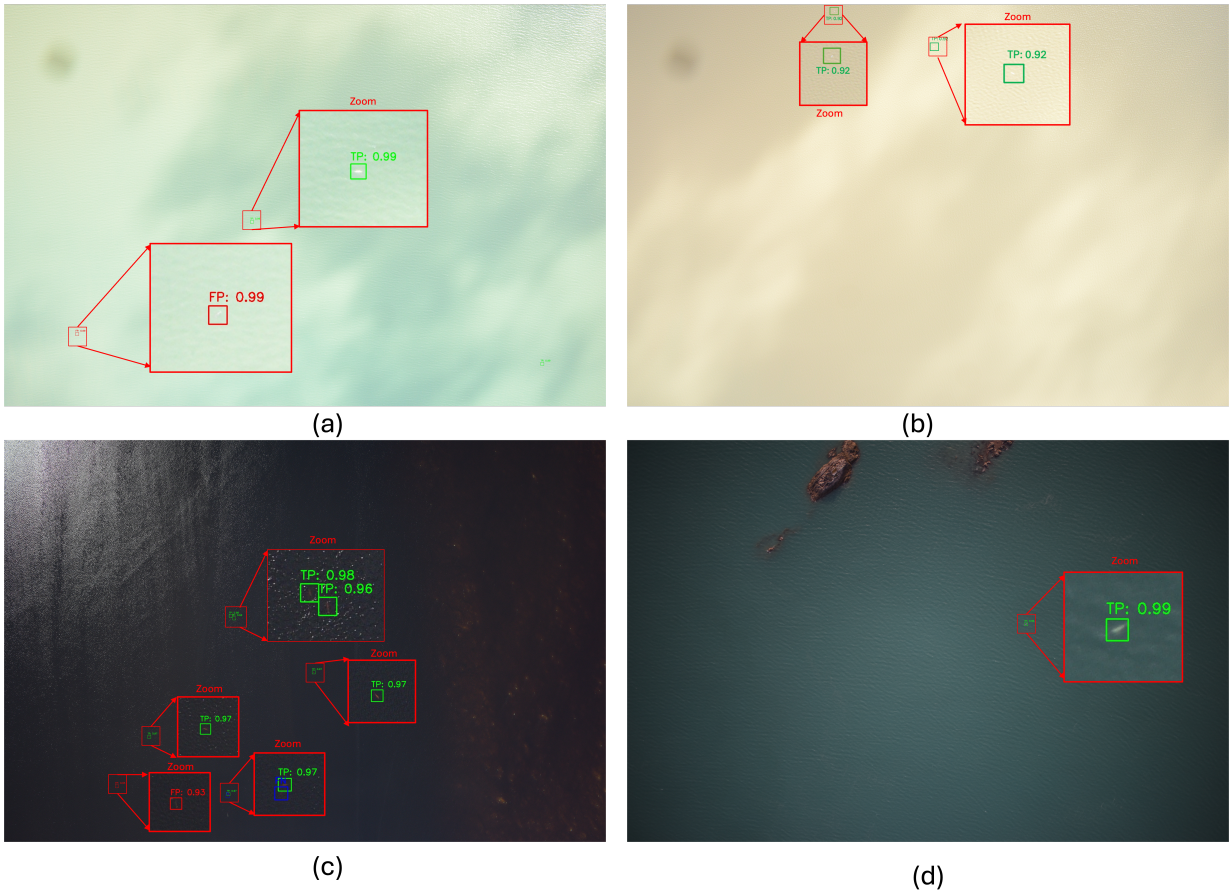


Figure 2.15: Prediction of FasterRCNN with ConvNext-T backbone on various backgrounds. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here.

To evaluate the model's performance in a real-world scenario, we applied it to the entire survey dataset comprising 39,492 images across all surveys. According to DFO's ground truth, 37,848 images are whale-free. Upon inferring our model to these images,

2,532 images were flagged as containing whale detections, with a total of 46,209 individual whale predictions.

Given the high number of detections, we investigated further by sorting images based on the number of predictions and manually reviewing those with more than five predicted whales. This examination revealed that 538 of these high-prediction images actually contained 42,521 valid whale detections, confirming the model's ability to accurately identify significant whale aggregations that might have been overlooked in earlier annotation by DFO. Fig. 2.16 (a) and (b) illustrates two examples of these overlooked whale images.

In summary, our model identified that out of the 37,848 images initially classified as whale-free, 538 images contain 42,521 valid whale detections, and 1,994 images contain 3,688 predictions. Although a comprehensive manual verification of these detections was not feasible due to time constraints, the model's efficiency in identifying images potentially containing whales represents a substantial improvement in the workflow for DFO experts. By reducing the number of images requiring expert review from 37,848 to 1,994, our model offers a significant reduction in manual analysis time.

DFO experts typically spend around 14 minutes annotating each image, which would amount to approximately 8,831 hours for all 37,848 images. With our model's assistance, the annotation time for the filtered 1,994 images is reduced to about 33 hours, where an expert spends only 1 minute per image due to model-assisted annotation (see section 3.0.3). This represents a reduction of approximately 267 times in time and costs, thereby accelerating the survey process and facilitating more frequent or extensive monitoring efforts.

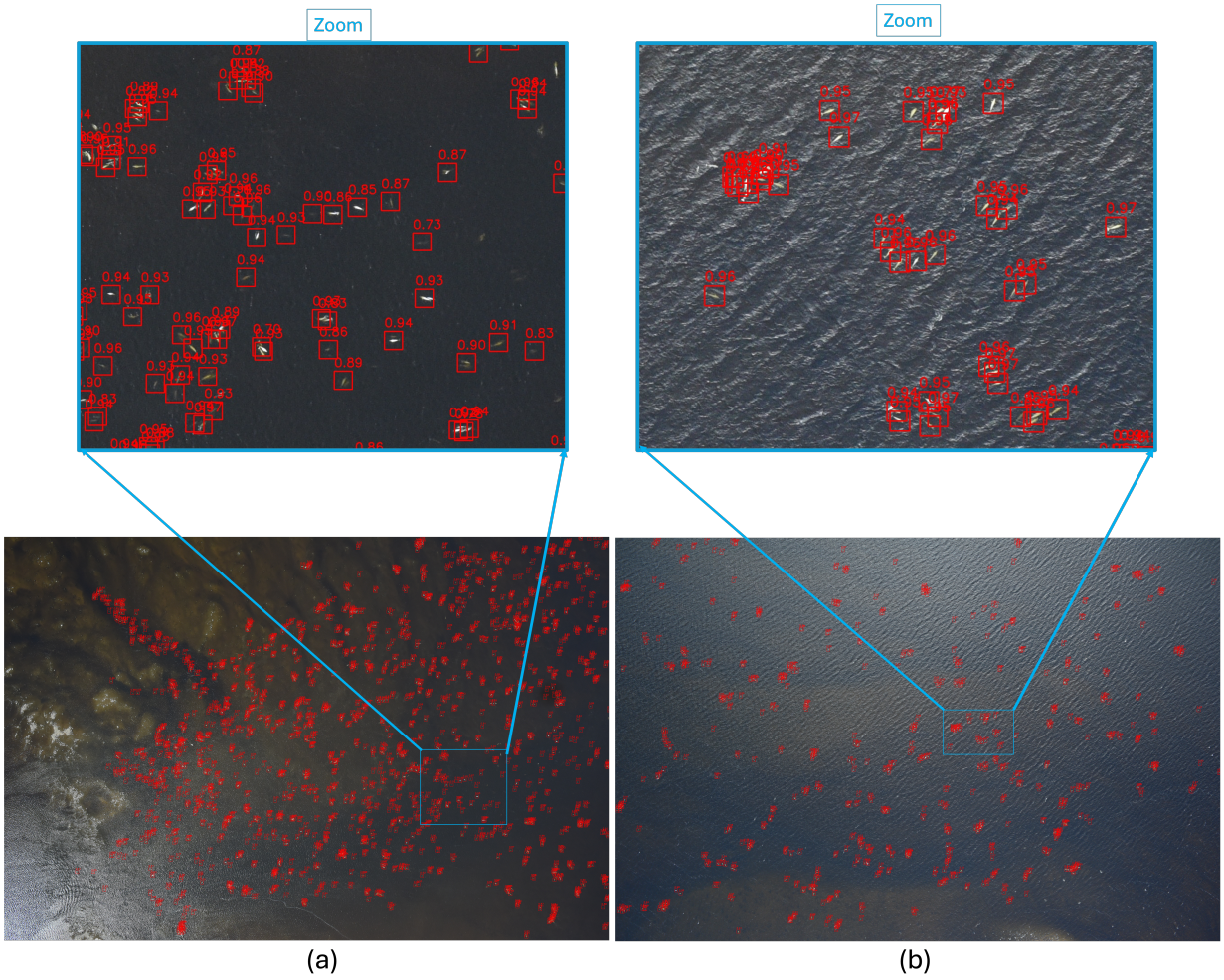


Figure 2.16: Predictions on images overlooked by DFO but found by our best model. Figure (a) contains around 2000 whales, and (b) contains around 865 whales. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here.

Chapter 3

Adapting to New Domain: Case Study

The DFO conducts annual surveys to assess marine mammal populations, often revisiting established locations or exploring new areas and species. In 2023, the DFO conducted a survey to update abundance estimates for narwhal, beluga, and bowhead whales in the Canadian High Arctic region(Fig. 3.1). This survey presents unique challenges for automated detection models due to the introduction of previously unencountered environmental conditions, particularly the presence of sea ice (Fig. 3.2).

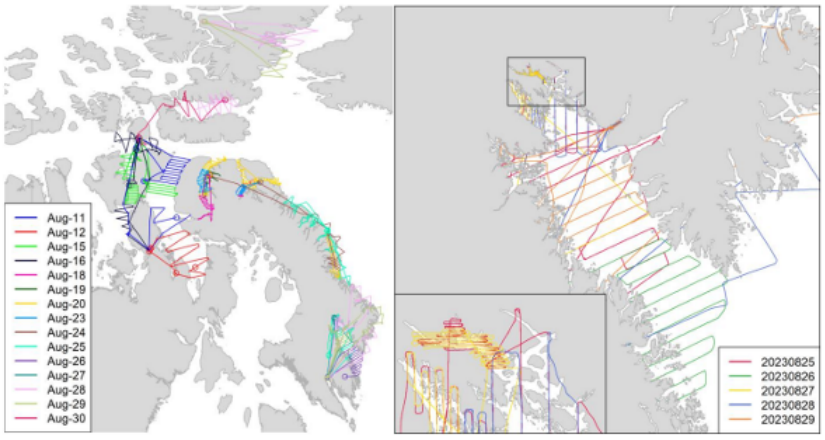


Figure 3.1: Map showing the survey transects over the Canadian High Arctic region.

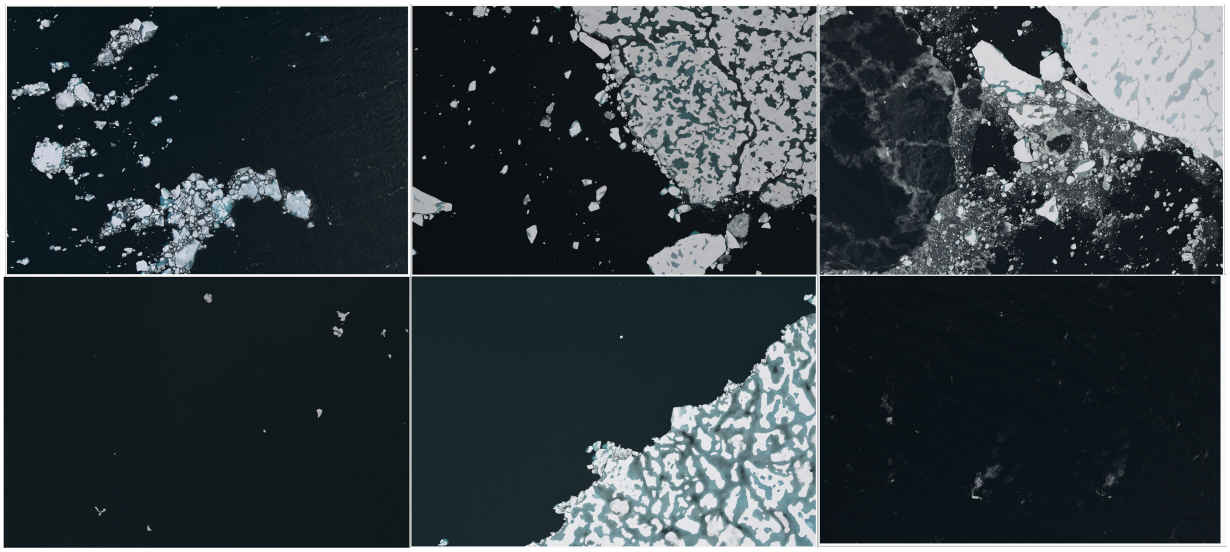


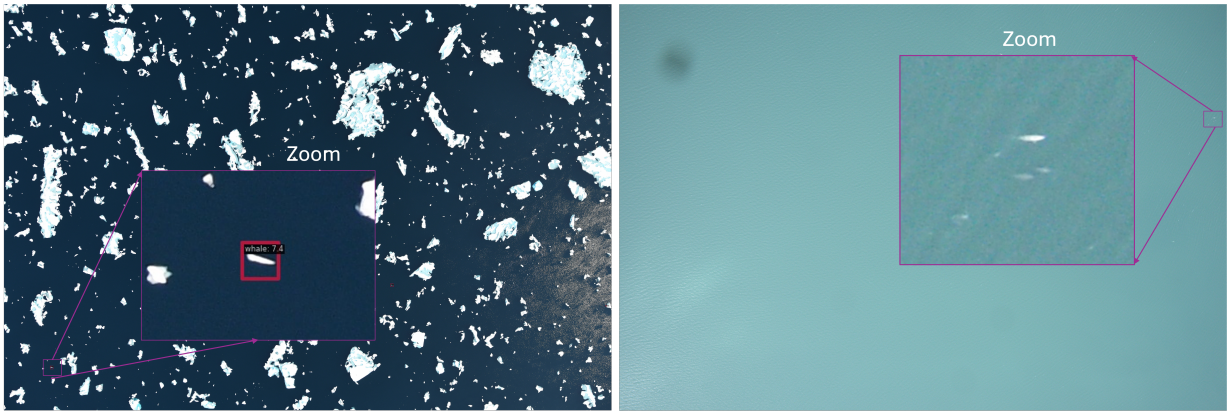
Figure 3.2: Icy conditions in the 2023 survey not present in earlier surveys.

The varying environmental conditions significantly influences the visual characteristics of the target species, creating obstacles for automated detection systems. Of particular note is the visual similarity between ice fragments and beluga whales in the aerial imagery (Fig. 3.3). This resemblance introduces a new dimension of complexity to the detection task, as the model must now differentiate between true cetacean targets and visually similar environmental features.

Our previous best model, a Faster R-CNN with a ConvNext-T backbone and 4096 patch size, demonstrated excellent performance on earlier datasets. However, when applied to the 2023 survey data, the model exhibited significant limitations. These included:

- Numerous false positives, likely due to misclassification of ice fragments as beluga whales.
- Reduced confidence in true positive detections (Fig. 3.5).
- Missed detections of species or morphologies not present in earlier surveys (Fig. 3.4).

Fig. 3.7 summarizes the key differences in appearance between the historical and new survey datasets, highlighting the challenges posed by this novel environment. The performance degradation observed in our model, when applied to the 2023 survey data, underscores the critical importance of domain adaptation in machine learning systems for



(a) 2023 Survey (b) 2014 CS Survey

Figure 3.3: The small ice shards in 2023 surveys closely resemble the beluga whales of the 2014 survey. Best viewed when zoomed in. Zoom in here.

wildlife monitoring. This challenge represents an active area of research in the field of computer vision and machine learning. To address this issue, we propose implementing an active learning approach.

Active learning is a semi-supervised machine learning paradigm that aims to achieve high accuracy with minimal labeled data by iteratively selecting the most informative samples for annotation (see Fig. 3.6). In the context of our cetacean detection challenge, active learning can efficiently identify and prioritize the most ambiguous or challenging images for expert annotation, thereby improving the model's performance in the new domain. The subsequent section elucidates our approach to designing an effective acquisition function, detailing the active learning pipeline, and describing its implementation within an integrated software environment called Label Studio. This pipeline is currently employed by the DFO to annotate their new survey data, representing a practical application of our research findings.

3.0.1 Designing acquisition function

Acquisition functions play a pivotal role in active learning by determining which unlabeled samples should be prioritized for annotation to maximize model improvement. The literature presents several effective approaches to this critical task. Uncertainty Sampling [36], a widely adopted method, capitalizes on the model's confidence scores to identify and select predictions with the least confidence or highest entropy. This approach effectively targets

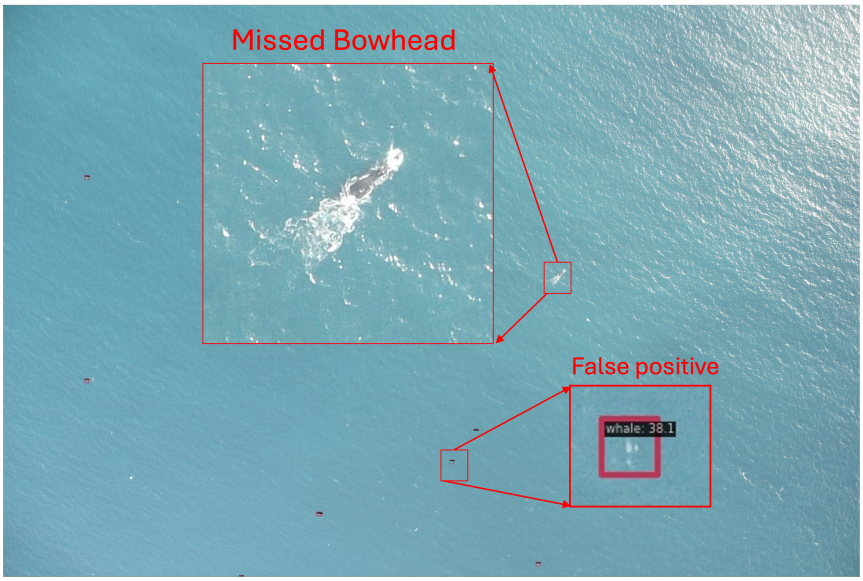


Figure 3.4: Bowhead whale which is not present in the earlier surveys missed by our model. Best viewed when zoomed in. Zoom in here.

the model's areas of greatest uncertainty.

Diversity-based methods[54], in contrast, aim to curate a representative subset of samples that encapsulates the variability present in the entire dataset. This approach ensures that the model is exposed to a wide range of data characteristics, potentially improving its generalization capabilities. Another prominent strategy is the Query-By-Committee (QBC) method[2]. This approach leverages an ensemble of models and identifies samples where there is maximum disagreement among the ensemble members. By focusing on these contentious samples, QBC aims to resolve areas of model uncertainty and improve overall performance.

These acquisition functions, among others, are widely used in various domains of machine learning, including computer vision, natural language processing, and bioinformatics[60]. In the context of cetacean detection from aerial imagery, we can leverage a critical prior to enhance the design of our acquisition function: the social behavior of whales, which typically move in pods.

To validate this prior, we analyzed the distribution of whales per image. Interestingly, the distribution closely aligns with the Pareto principle[52], commonly known as the 80-20 rule. As illustrated in Fig. 3.8, approximately 20% of the images contain 80% of the total whale observations, mirroring the principle's application in wealth distribution.

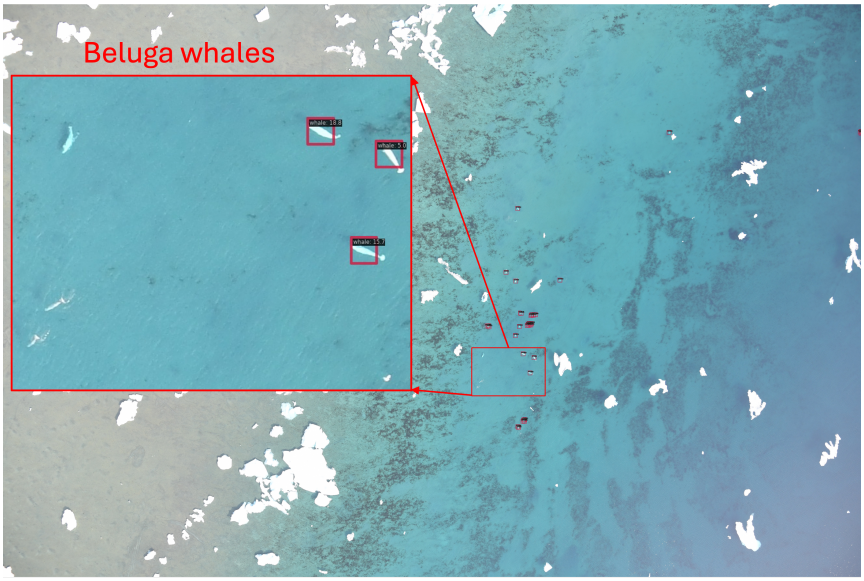


Figure 3.5: Beluga whales detected with low confidence along with some clear whales not predicted, using our best model without any fine-tuning. Best viewed when zoomed in. Zoom in here.

We integrated this prior knowledge with an uncertainty-based sampling approach. Our chosen uncertainty metric is the average confidence score per image, mathematically expressed as:

$$\varphi^i_{avg} = \frac{1}{\left|\mathcal{S}^\theta_i\right|} \sum_{j \in \mathcal{S}^\theta_i} \hat{\mathbf{c}}_j$$

where $\mathcal{S}^\theta_i$ is the set of boxes with score $\geq \theta$
and $\left|\mathcal{S}^\theta_i\right|$ denotes the number of objects in $\mathcal{S}^\theta_i$
and $\hat{\mathbf{c}}_j$ is the jth prediction in the image.

High values of φ^i_{avg} indicate images likely to contain objects in which the model exhibits high confidence. Prioritizing such images for training is particularly advantageous given the significant class imbalance in aerial surveys, where only approximately 4% of the images contain whales. Fig. 3.9 illustrates our novel portfolio-based acquisition function designed for aerial object detection. This approach strategically selects images with varying object densities, applying different weightages to ensure a balanced exploration of the search space.

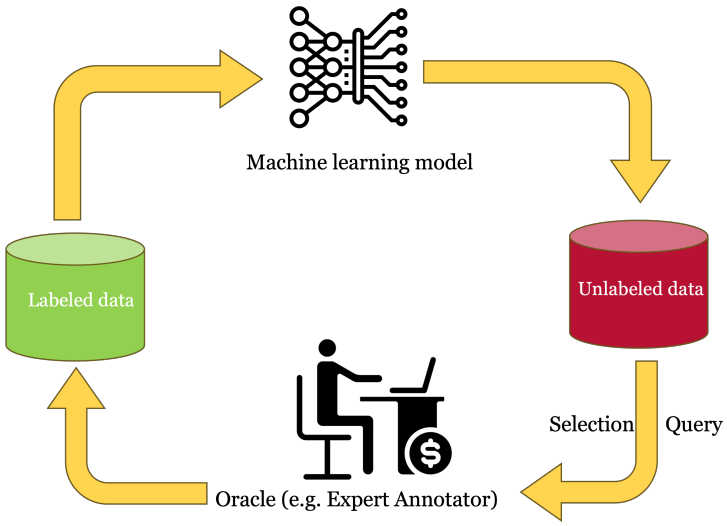


Figure 3.6: Overview of active learning paradigm.

Our acquisition strategy involves first categorizing images into five bins based on the number of detected objects. Within each bin, images are then sorted according to their φ^i_{avg} values. This dual-criteria approach allows us to balance between selecting potentially whale-rich images and images where the model shows varying levels of certainty (promoting model improvement across different scenarios). This structured sampling ensures comprehensive coverage of various scenarios while focusing more resources on potentially whale-rich images, thereby optimizing the efficiency of the annotation process and subsequent model training.

3.0.2 Implementation: Active Learning Workflow

The successful implementation of an active learning strategy for cetacean detection necessitates an integrated software solution that encompasses all components of the process, from image querying to model fine-tuning. To achieve this, we leveraged Label Studio[1], an open-source tool that offers seamless integration between annotation and model training phases. Label Studio provides several key advantages for our workflow:

- Integration Capabilities: Label Studio allows for smooth integration with our custom acquisition function and model training pipeline.

[1]https://labelstud.io/

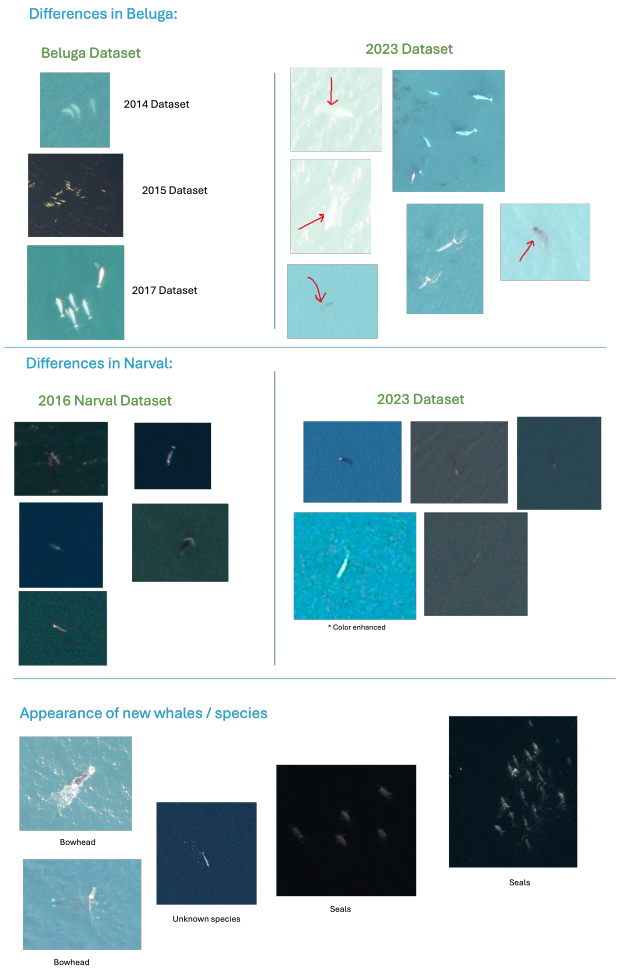


Figure 3.7: Summary of differences in appearance between images from old and new survey.

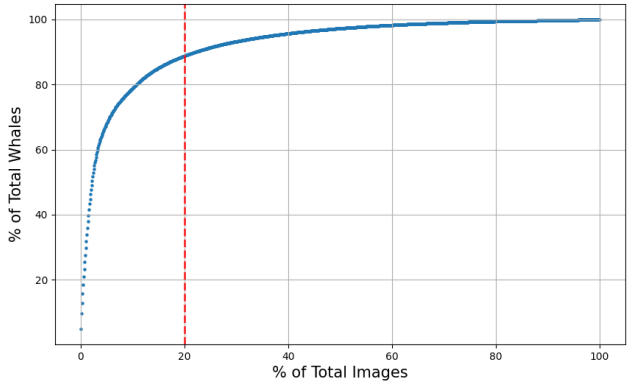


Figure 3.8: Cummulative distribution of whales per image from old surveys sorted in descending order. 20% of images owns 80% of total whales.

- AI-Assisted Annotation: The platform incorporates model predictions to expedite the annotation process, significantly reducing the manual effort required from experts.
- User-Friendly Interface: Annotators can easily add, edit, or delete model predictions through an intuitive interface, ensuring high-quality ground truth data.

Fig. 3.10 illustrates our active learning workflow implemented using Label Studio. For the fine-tuning stage, we keep 80% of images for training and 20% of images for testing. Fig. 3.11 shows the annotation window of Label Studio.

3.0.3 Result

This section presents the results of our active learning pipeline, highlighting the outcomes before and after fine-tuning across four iterations (400 images total). In each iteration, 100 images were selected and annotated by DFO experts using Label Studio, with model predictions provided as a reference during annotation. The average annotation time per image, as recorded in Label Studio, was 60 seconds. After annotation, in each iteration, 80 images were allocated for training, while 20 were reserved for testing. To maintain data integrity and prevent data leakage, performance metrics are reported exclusively on the test images set aside during the fine-tuning phase.

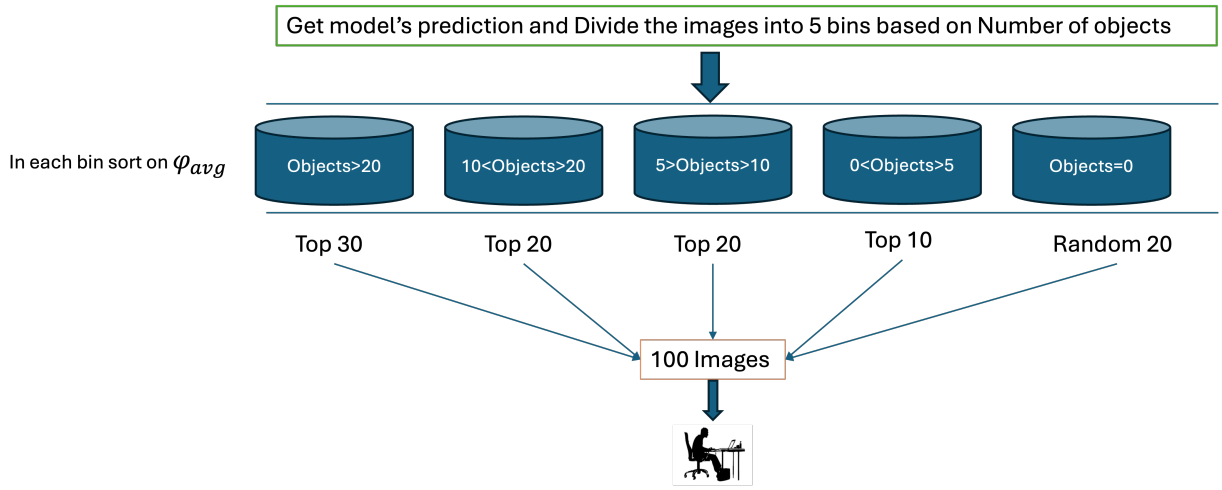


Figure 3.9: Portfolio-based acquisition function

Table 3.1: Performance metrics across four active learning iterations, showing the Average Precision at IoU=10 ($AP_{IoU=10}$), True Positives (TP), False Positives (FP), and False Negatives (FN). The metrics are reported on the test set images after fine-tuning on the corresponding training images.

AL iter	$AP_{IoU=10}$	**TP**	**FP**	**FN**
1	0.601	16	21	1
2	0.775	24	9	4
3	0.730	230	83	9
4	0.740	10	4	5
Total		280	117	19

Table 3.1 provides a detailed breakdown of the results across each active learning iteration, including $AP_{IoU=10}$, as well as the number of True Positives (TP), False Positives (FP), and False Negatives (FN).

Fig. 3.12 shows an image containing fewer than five objects. In the previous dataset, points were converted to square bounding boxes as annotations or GT, while in the new survey, the analyst directly annotated with rectangles. As a result, the fine-tuned model (see Fig. 3.12(a)) has adapted to rectangular-shaped boxes.

Fig. 3.12 shows an image with more than 20 objects acquired using our active learning pipeline. Notice how the fine-tuned model has detected whales that were previously missed. Although the model is not yet perfect and requires a few more iterations, detection has

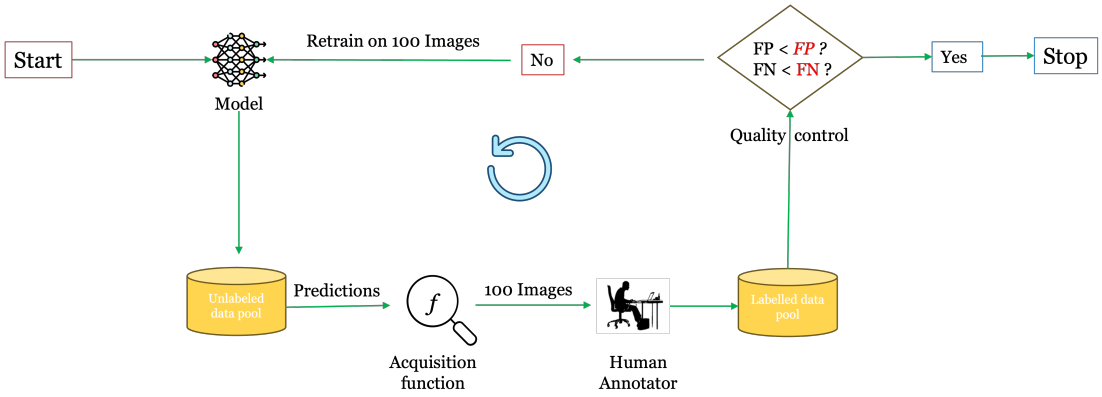


Figure 3.10: Active Learning Workflow implemented using Label Studio

improved.

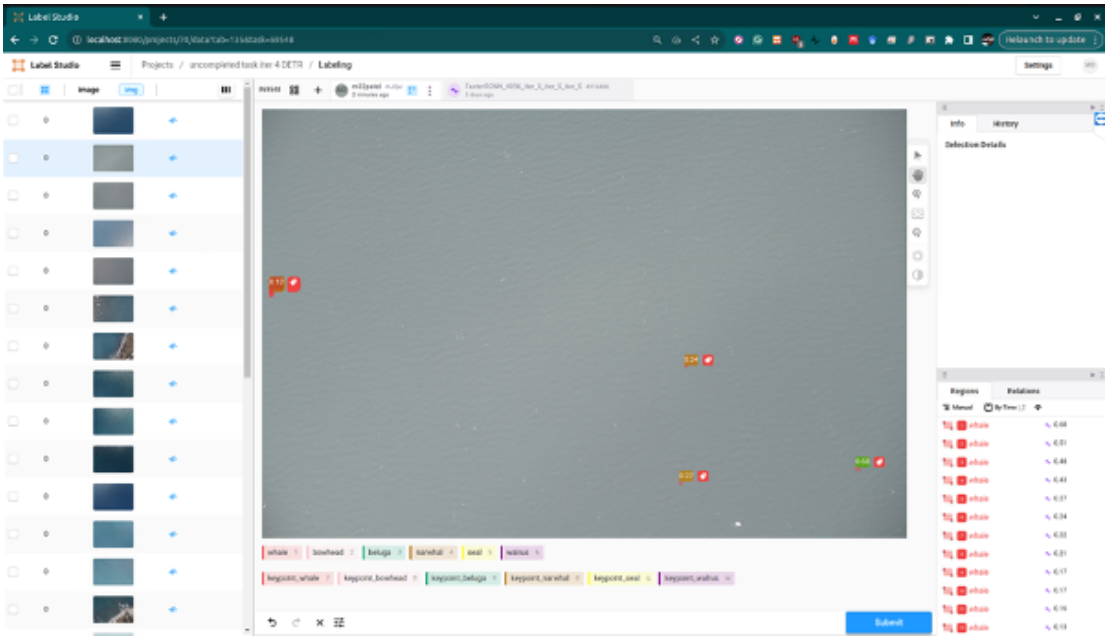
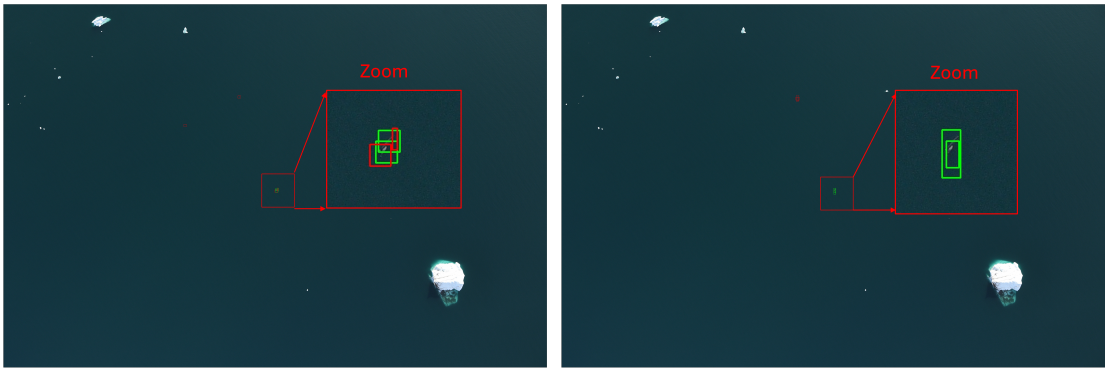


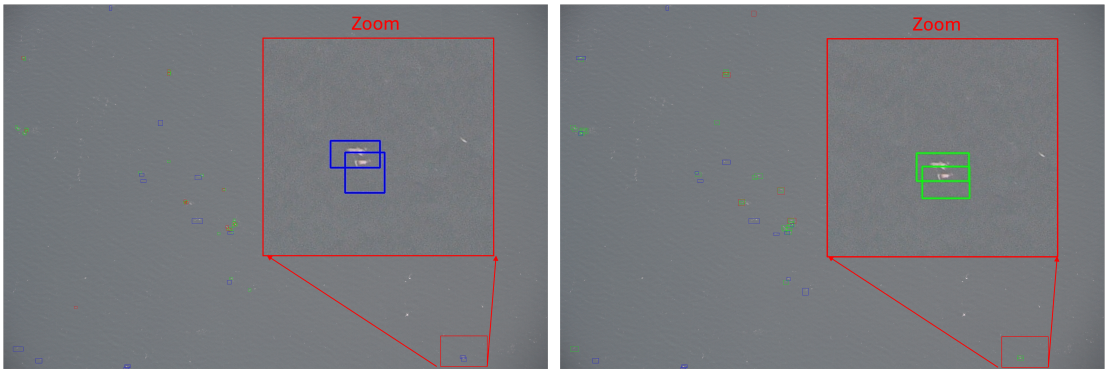
Figure 3.11: The labelstudio annotation interface.



(a) Model's prediction before active learning loop

(b) Model's prediction after active learning loop

Figure 3.12: Image with fewer than five objects acquired using our acquisition function. (a) shows the model's predictions before fine-tuning. (b) shows the predictions after fine-tuning. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in.Zoom in here.



(a) Model's prediction before active learning loop

(b) Model's prediction after active learning loop

Figure 3.13: Image with more than 20 objects acquired using our acquisition function. (a) shows the model's predictions before fine-tuning. (b) shows the predictions after fine-tuning. Box color coding: Red - FP, Blue - FN, Green - TP. The number above the boxes represents the confidence score of the model. Best viewed when zoomed in. Zoom in here.

Chapter 4

Conclusion

4.1 Conclusion

In this study, we present a comprehensive solution for whale detection, addressing key challenges in marine mammal monitoring through aerial surveys. We first present a comparative analysis of various object detection architectures and common vision backbones, with a specific focus on their performance in detecting small objects such as whales. Our research provides the first comprehensive examination of window size effects on detection performance. We demonstrate that larger window sizes generally enhance the contextual understanding of models, leading to improved detection performance. FasterRCCN with ConvNext-T backbone trained on patch size 4096 enabled us to achieve a remarkable $AP_{IoU=10}$ of 0.878, setting a new benchmark in whale detection. Additionally, our model excels in all-negative image precision, a crucial metric for reducing false alarms and significantly decreasing the workload associated with eliminating false positives in semi-automated deep learning systems. By narrowing down the images requiring expert review from 39,492 to 3,638, our model offers a considerable reduction in manual analysis time.

To ensure the long-term utility of our model, we integrate an active learning framework, enabling adaptation to domain shifts - a common challenge in real-world applications. We develop a novel portfolio-based acquisition function specifically tailored for aerial surveys, leveraging the social behavior of whales. Through a case study, we demonstrate the effectiveness of this proposed function.

The impact of our research is multifaceted: it enhances detection accuracy, reduces manual workload, ensures adaptability to changing environments, and provides a tailored

approach for aerial surveys. By offering a more accurate, efficient, and adaptable whale detection system, our research significantly contributes to marine conservation efforts and facilitates our understanding of whale populations and behaviors.

In the context of climate change, where sea ice is melting and Arctic routes are opening, our model proves to be particularly valuable. Fig. 4.1 shows an image of a whale from the 2023 survey detected near a ship. Thus our model can be used to locate whales before planning routes in these newly accessible areas, minimizing the risk of ship-whale collisions and supporting safer navigation in vulnerable marine environments.

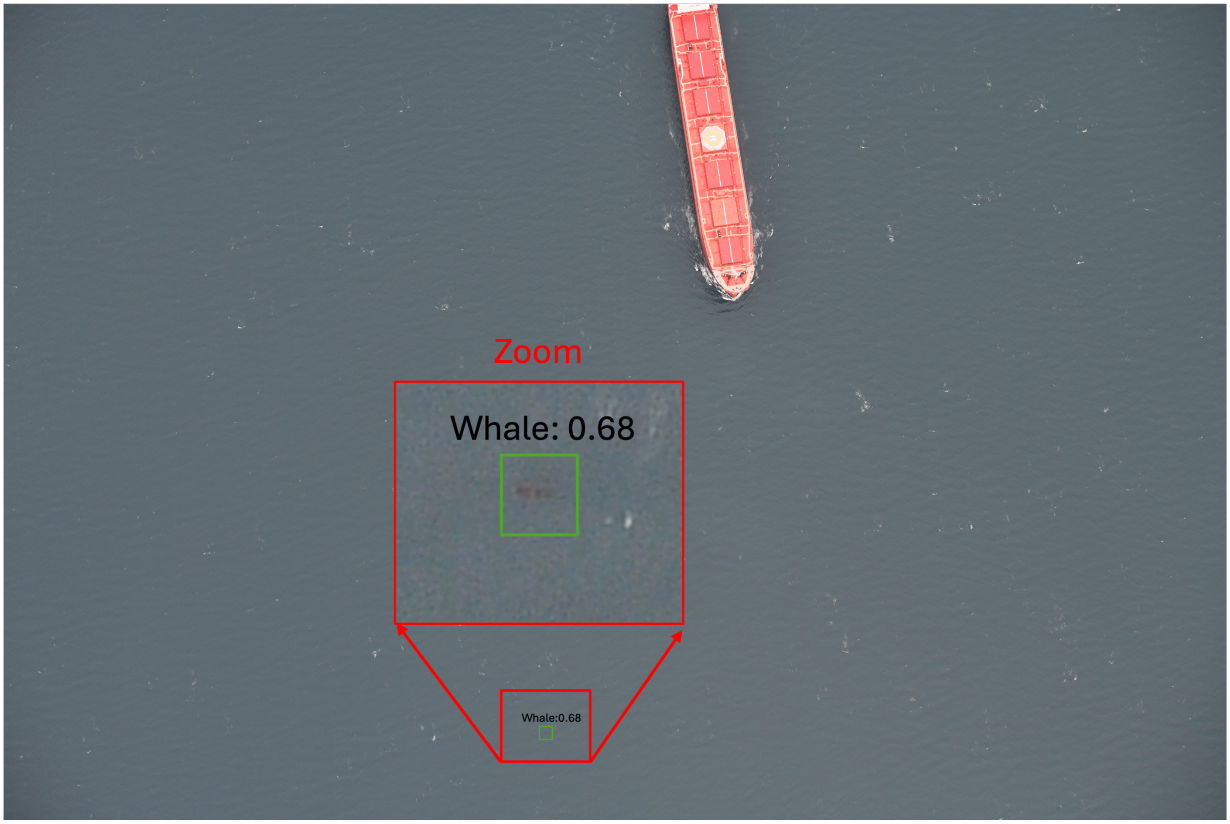


Figure 4.1: A whale is detected near the path of a ship

In conclusion, this study not only advances the technical aspects of whale detection but also provides a practical, adaptable tool for marine researchers and conservationists, representing a significant step forward in the field of marine mammal monitoring and conservation.

4.2 Limitations and future work

While our study presents significant advancements in whale detection, several limitations and areas for future research have been identified:

1. Multi-class Whale Classification: Our current model focuses solely on single-category whale detection. A more comprehensive approach would involve developing a model capable of classifying whales into categories such as calf, juvenile, and adult. This multi-class classification would provide more detailed information for marine biologists and conservationists, enhancing our understanding of whale population dynamics. We propose this as a key area for future research.

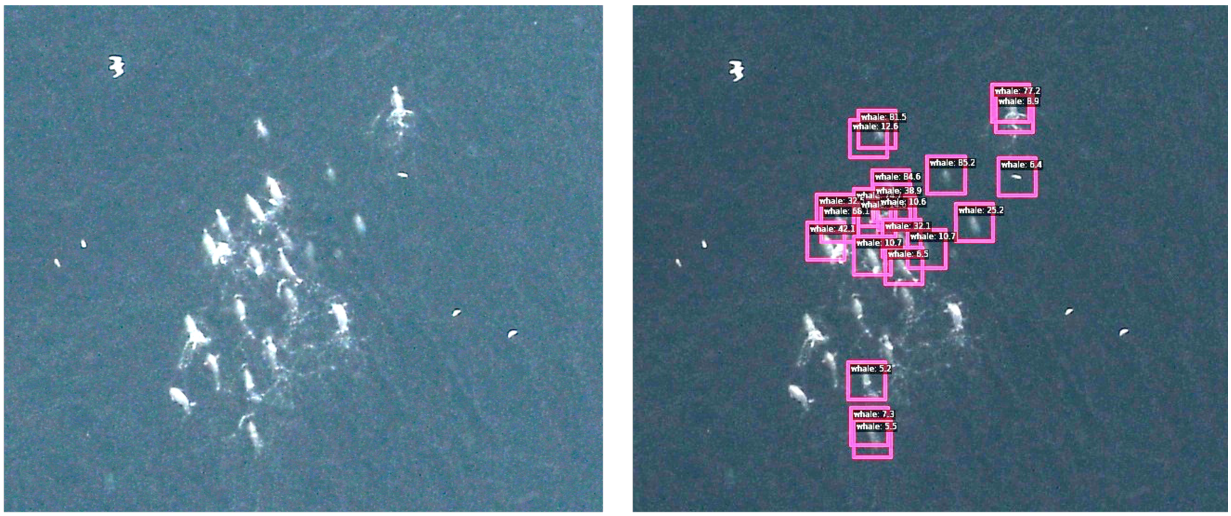
2. Enhanced Active Learning Strategies: Our active learning approach, while effective, could be further improved by incorporating geographical data. At the time of this study, geocoded image data was not available, limiting our ability to implement distance-based sampling techniques. Future work should explore the integration of geographical distance-based sampling, leveraging the spatial clustering tendency of whales. We envision a multi-agent image exploration approach where: a) Sampling prioritizes images adjacent to known whale sightings. b) Multiple "samplers" explore different regions of the survey area concurrently. c) Agents communicate discoveries to influence each other's search patterns. d) The system dynamically adjusts the exploration-exploitation balance as more whale sightings are confirmed. This strategy could significantly enhance the efficiency of identifying whale-rich areas and improve model performance across diverse geographical contexts.

3. Open-Vocabulary Wildlife Detection: Our model's unexpected ability to detect seals without specific training(see Fig. 4.2) highlights the potential for a more generalized wildlife detection approach. With the increasing availability of diverse species datasets, there is an opportunity to develop a foundational wildlife detection model for aerial imagery. Future research should explore the implementation of models like Grounding-DINO[37] to support open-vocabulary detection. This approach could leverage the shared features of wildlife in aerial imagery, potentially creating a more versatile and robust detection system applicable to various species.

4. Using Satellite Imagery: To align our research with other ongoing whale detection efforts, such as those explored by the Department of Fisheries and Oceans (DFO) and initiatives like the SmartWhales[1] space detection system, our approach could serve as

[1] https://www.asc-csa.gc.ca/eng/blog/2024/06/04/smartwhales-using-space-to-keep-whales-safe.asp

a complementary tool. The integration of our whale detection model with satellite-based systems like SmartWhales could enhance detection accuracy by combining high-resolution aerial imagery with broader spatial coverage from satellites. This synergy could lead to more comprehensive monitoring solutions, allowing for more frequent and better tracking of whale populations.



(a) Seals in 2023 Survey

(b) Seals detected as whales by our best model without specific training

Figure 4.2: Seals detected by our best model without any finetuning on the new 2023 survey.

By addressing these limitations and pursuing these research directions, we aim to further advance the field of marine wildlife detection and monitoring. These enhancements would not only improve the accuracy and versatility of our whale detection system but also contribute to broader wildlife conservation efforts through more comprehensive and adaptable detection technologies.

References

[1] Hüseyin Gökhan Akçay, Bekir Kabasakal, Duygugül Aksu, Nusret Demir, Melih Öz, and Ali Erdoğan. Automated bird counting with deep learning for regional bird distribution mapping. *Animals*, 10(7):1207, 2020.

[2] Shlomo Argamon-Engelson and Ido Dagan. Committee-based sample selection for probabilistic classifiers. *Journal of Artificial Intelligence Research*, 11:335–360, 1999.

[3] CCG Bamford, N Kelly, L Dalla Rosa, DE Cade, PT Fretwell, PN Trathan, HC Cubaynes, AFC Mesquita, L Gerrish, AS Friedlaender, et al. A comparison of baleen whale density estimates derived from overlapping satellite imagery and a shipborne survey. *Scientific reports*, 10(1):12985, 2020.

[4] Justine Boulent, Bertrand Charry, Malcolm McHugh Kennedy, Emily Tissier, Raina Fan, Marianne Marcoux, Cortney A Watt, and Antoine Gagné-Turcotte. Scaling whale monitoring using deep learning: A human-in-the-loop solution for analyzing aerial datasets. *Frontiers in Marine Science*, 10:1099479, 2023.

[5] Alexander Buslaev, Vladimir I. Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A. Kalinin. Albumentations: Fast and flexible image augmentations. *Information*, 11(2):125, February 2020.

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[7] Kelly Cates, Doug P Demaster, RL Brownell Jr, Gregory Silber, Scott Gende, Russell Leaper, Fabian Ritter, and Simon Panigada. Strategic plan to mitigate the impacts of ship strikes on cetacean populations: 2017-2020. *IWC*, 2017.

[8] Dominique Chabot and Charles M Francis. Computer-automated bird detection and counts in high-resolution aerial images: a review. *Journal of Field Ornithology*, 87(4):343–359, 2016.

[9] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[10] Alexandre Delplanque, Samuel Foucher, Jérôme Théau, Elsa Bussière, Cédric Vermeulen, and Philippe Lejeune. From crowd to herd counting: How to precisely detect and count african mammals using aerial imagery and deep learning? *ISPRS Journal of Photogrammetry and Remote Sensing*, 197:167–180, 2023.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[12] Camille Desjonquères, Sara Villén-Pérez, Paulo De Marco, Rafael Márquez, Juan F Beltrán, and Diego Llusia. Acoustic species distribution models (asdms): A framework to forecast shifts in calling behaviour under climate change. *Methods in Ecology and Evolution*, 13(10):2275–2288, 2022.

[13] Antoine M Dujon, Daniel Ierodiaconou, Johanna J Geeson, John PY Arnould, Blake M Allan, Kostas A Katselidis, and Gail Schofield. Machine learning to detect marine animals in UAV imagery: Effect of morphology, spacing, behaviour and habitat. *Remote Sensing in Ecology and Conservation*, 7(3):341–354, 2021.

[14] Mahdi Esfahanian. *Detection and classification of marine mammal sounds*. Florida Atlantic University, 2014.

[15] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[16] Bento C Gonçalves, Michael Wethington, and Heather J Lynch. Sealnet 2.0: human-level fully-automated pack-ice seal detection in very-high-resolution satellite imagery with cnn model ensembles. *Remote Sensing*, 14(22):5655, 2022.

[17] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1994.

[18] Katherine M Green, Mala K Virdee, Hannah C Cubaynes, Angelica I Aviles-Rivero, Peter T Fretwell, Patrick C Gray, David W Johnston, Carola-Bibiane Schönlieb, Leigh G Torres, and Jennifer A Jackson. Gray whale detection in satellite imagery using deep learning. *Remote Sensing in Ecology and Conservation*, 9(6):829–840, 2023.

[19] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[20] Emilio Guirado, Siham Tabik, Marga L Rivas, Domingo Alcaraz-Segura, and Francisco Herrera. Whale counting in satellite and aerial images with deep learning. *Scientific reports*, 9(1):14259, 2019.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[22] Peiyuan Jiang, Daji Ergu, Fangyao Liu, Ying Cai, and Bo Ma. A review of yolo algorithm developments. *Procedia computer science*, 199:1066–1073, 2022.

[23] Thani Jintasuttisak, Andrew Leonce, Moayyed Sher Shah, Tamer Khafaga, Greg Simkins, and Eran Edirisinghe. Deep learning based animal detection and tracking in drone video footage. ICCAI '22, page 425–431, New York, NY, USA, 2022. Association for Computing Machinery.

[24] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8, 2023.

[25] Amanda Johnson, Glenn Salvador, John Kenney, Jooke Robbins, Scott Kraus, Scott Landry, and Phil Clapham. Fishing gear involved in entanglements of right and humpback whales. *Marine Mammal Science*, 21(4):635–645, 2005.

[26] Donald Knuth. *The TeXbook*. Addison-Wesley, Reading, Massachusetts, 1986.

[27] Peter Kulits, Jake Wall, Anka Bedetti, Michelle Henley, and Sara Beery. Elephantbook: A semi-automated human-in-the-loop system for elephant re-identification. In *Proceedings of the 4th ACM SIGCAS Conference on Computing and Sustainable Societies*, pages 88–98, 2021.

[28] Leslie Lamport. *LaTeX — A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, second edition, 1994.

[29] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[30] Peter Q Lee, Keerthijan Radhakrishnan, David A Clausi, K Andrea Scott, Linlin Xu, and Marianne Marcoux. Beluga whale detection in the cumberland sound bay using convolutional neural networks. *Canadian Journal of Remote Sensing*, 47(2):276–294, 2021.

[31] Yuexiang Li, Xinpeng Xie, Linlin Shen, and Shaoxiong Liu. Reverse active learning based atrous densenet for pathological image classification. *BMC bioinformatics*, 20:1–15, 2019.

[32] Yuxuan Li, Qibin Hou, Zhaohui Zheng, Ming-Ming Cheng, Jian Yang, and Xiang Li. Large selective kernel network for remote sensing object detection. *arXiv preprint arXiv:2303.09030*, 2023.

[33] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[34] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

[35] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128:261–318, 2020.

[36] Shang Liu and Xiaocheng Li. Understanding uncertainty sampling. *arXiv preprint arXiv:2307.02719*, 2023.

[37] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[38] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018.

[39] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Computer*

Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pages 21–37. Springer, 2016.

[40] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model, 2024.

[41] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

[42] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022.

[43] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[44] Tao Lu, Baokun Han, and Fanqianhui Yu. Detection and classification of marine mammal sounds using alexnet with transfer learning. *Ecological Informatics*, 62:101277, 2021.

[45] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29, 2016.

[46] David K Mellinger, Kathleen M Stafford, Sue E Moore, Robert P Dziak, and Haru Matsumoto. An overview of fixed passive acoustic observation methods for cetaceans. *Oceanography*, 20(4):36–45, 2007.

[47] Rachelle Nadaf. Five ways researchers find whales in the open ocean.

[48] Alexander Neubeck and Luc Van Gool. Efficient non-maximum suppression. In *18th international conference on pattern recognition (ICPR'06)*, volume 3, pages 850–855. IEEE, 2006.

[49] Gregory M O'Corry-Crowe. Beluga whale: Delphinapterus leucas. In *Encyclopedia of marine mammals*, pages 108–112. Elsevier, 2009.

[50] Muhammed Patel, Xinwei Chen, Linlin Xu, Fernando J. Pena Cantu, Javier Noa Turnes, Neil C. Brubacher, David A. Clausi, and K. Andrea Scott. The influence

of input image scale on deep learning-based beluga whale detection from aerial remote sensing imagery. In *IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*, pages 5732–5734, 2023.

[51] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[52] Robert Sanders. The pareto principle: its use and abuse. *Journal of Services Marketing*, 1(2):37–40, 1987.

[53] Iqbal H Sarker. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. *SN computer science*, 2(6):420, 2021.

[54] Dan Shen, Jie Zhang, Jian Su, Guodong Zhou, and Chew Lim Tan. Multi-criteria-based active learning for named entity recognition. In *Proceedings of the 42nd annual meeting of the Association for Computational Linguistics (ACL-04)*, pages 589–596, 2004.

[55] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[56] STEVEN L Swartz, Tim Cole, MARK A McDonald, JOHN A Hildebrand, Erin M Oleson, ANTHONY Martinez, PHILLIP J Clapham, Jay Barlow, and M Lou Jones. Acoustic and visual survey of humpback whale (megaptera novaeangliae) distribution in the eastern and southeastern caribbean sea. *Caribbean Journal of Science*, 39(2):195–208, 2003.

[57] AA Tyshko, MA Krinitskiy, AV Shatravin, and RA Belikov. Automatic detection of acoustic signals of beluga whales and bottlenose dolphins. *Moscow University Physics Bulletin*, 78(Suppl 1):S217–S225, 2023.

[58] Zeyu Xu, Tiejun Wang, Andrew K Skidmore, and Richard Lamprey. A review of deep learning techniques for detecting animals in aerial and satellite images. *International Journal of Applied Earth Observation and Geoinformation*, 128:103732, 2024.

[59] Xiaoyuan Yu, Jiangping Wang, Roland Kays, Patrick A Jansen, Tianjiang Wang, and Thomas Huang. Automated identification of animal species in camera trap images. *EURASIP Journal on Image and Video Processing*, 2013:1–10, 2013.

[60] Xueying Zhan, Qingzhong Wang, Kuan-hao Huang, Haoyi Xiong, Dejing Dou, and Antoni B Chan. A comparative survey of deep active learning. *arXiv preprint arXiv:2203.13450*, 2022.

[61] Wentao Zhu, Can Zhao, Wenqi Li, Holger Roth, Ziyue Xu, and Daguang Xu. Lamp: Large deep nets with automated model parallelism for image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 374–384. Springer, 2020.

[62] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.