Making use of unlabeled data: Comparing strategies for marine animal detection in long-tailed datasets using self-supervised and semi-supervised pre-training

Tarun Sharma
California Institute of Technology, USA
tsharma@caltech.edu

Danelle E. Cline
MBARI, USA
dcline@mbari.org

Duane Edgington
MBARI, USA
duane@mbari.org

Abstract

This paper discusses strategies for object detection in marine images from a practitioner's perspective working with real-world long-tail distributed datasets with a large amount of additional unlabeled data on hand. The paper discusses the benefits of separating the localization and classification stages, making the case for robustness in localization through the amalgamation of additional datasets inspired by a widely used approach by practitioners in the camera-trap literature. For the classification stage, the paper compares strategies to use additional unlabeled data, comparing supervised, supervised iteratively, self-supervised, and semi-supervised pre-training approaches. Our findings reveal that semi-supervised pre-training, followed by supervised fine-tuning, yields a significantly improved balanced performance across the long-tail distribution, albeit occasionally with a trade-off in overall accuracy. These insights are validated through experiments on two real-world long-tailed underwater datasets collected by the Monterey Bay Aquarium Research Institute (MBARI).

1. Introduction

With the rise of the blue economy, studying ocean community composition and their ecosystems is essential to understanding the ecological impact activities like offshore energy and deep-sea mining will have on them. Oceanographic institutes have been surveying parts of the deep oceans using underwater vehicles fitted with video monitoring capabilities for many years now, resulting in a data deluge. Automating the analysis of this video data for biodiversity monitoring using supervised computer vision techniques like object detection has been successful in both ocean and land realms [9, 23]. This approach however requires expensive manual data annotations by taxonomists, localizing and categorizing every animal in a frame. While these annotations are crucial for our ability to automate video analysis to any extent, this results in the bulk of collected data, the unlabeled data, being completely unused for model training. Moreover, supervised computer vision models exhibit suboptimal performance on imbalanced datasets, particularly struggling with the accurate classification of rare entities. Given that datasets procured from natural environments invariably exhibit a long-tailed distribution, the shortcomings of these models become more pronounced. The erroneous identification of rare species during investigations assessing the ecological impact on oceanic communities holds the potential for significant repercussions. Consequently, an optimal objective entails achieving a balanced performance across all classes within the model's purview.

Self-supervised learning, a machine learning paradigm where a model is trained using implicit labels arising from inherent structures or relationships within the input data alone as a supervisory signal, has emerged as a viable strategy for leveraging unlabeled data. These methods, demonstrate in certain instances superior performance across various downstream tasks in comparison to their supervised counterparts [10, 13], have been shown to scale well with both data and model size [10], and are proficient few-shot learners particularly when trained with extensive corpora of unlabeled data [11]. Notably, these methods are more robust toward unbalanced datasets [17], hypothesized to result from a more uniform representation space [16]. The efficacy of self-supervised learning is most pronounced when unlabeled data originates from the same domain, the availability of supervised data is limited, and the task granularity is relatively coarse [8]. In the context of automated analysis of deep-sea video footage, this methodology aligns with at least two of these criteria.

Self-supervised learning can be broadly categorized into two main types, task-based methods and contrastive methods [5]. Task-based methodologies involve training models to perform a task, such as predicting the color composition [25], sequential order of image components [19], or even masked-out regions of an image [21]. Contrastive learning methods minimize the distance in the representation space between two semantically similar images, forming a positive pair, while maximizing the distance between two semantically dissimilar images, forming a negative pair. In the absence of explicit labels, positive pairs often consist of two augmented versions of the same image. One prominent instance of contrastive learning is SimCLR [7], wherein the authors show that the type of data augmentations used is a crucial factor affecting performance. Semi-supervised learning encompasses approaches to train models using a combination of labeled and unlabeled data. A typical semi-supervised strategy is to perform contrastive learning on unlabeled data, also called contrastive pre-training, followed by supervised fine-tuning using labeled data. Recent advances in semi-supervised learning have demonstrated enhanced performance and computational efficiency by using a small subset of labeled data during the contrastive pre-training step. This is done by the addition of an extra term in the loss function, such as the SuNCEt loss [3], or by assigning and subsequently minimizing the cross entropy loss of pseudo labels assigned to unlabeled data using a small set of supervised support samples as demonstrated in the PAWS method [4].

Strategies for leveraging unlabeled data to enhance biodiversity monitoring include task-based approaches, like ranking pairs (original image and a crop from the image) of unlabeled noisy sonar images based on the number of fish in them while simultaneously predicting the density maps of fish in a supervised manner using a subset of labeled images [22], pseudo labeling of unlabeled data using a supervised model [18], and contrastive learning approaches, like selecting positive pairs of images based on temporal or contextual relatedness from camera trap data as opposed to the standard approach of using two augmented versions of the same image [20]. Most contrastive or task-based pre-training approaches are focused on improving classification performance. Self-supervised object detection pre-training methods, wherein both the region proposal and classification heads are pre-trained, result in only limited enhancements compared to traditional object detection [14], often demand substantial computational resources, and do not effectively address the open-world problem. Conversely, a more straightforward localization approach, exemplified by the MegaDetector [6], a standard object detector model trained for animal localization in camera trap images, proves robust to unseen data and finds widespread use among non-profits and ecologists globally [2]. The MegaDetector's robustness and practical utility stem from its training on multiple datasets, made possible by reducing all classes into three overarching categories: 'animal,' 'vehicle,' and 'human.' This model, characterized by its simplicity, low computational cost, ease of fine-tuning, and adept handling of the open-world problem, successfully localizes previously unseen animals in novel backgrounds. Inspired by the effectiveness of this approach, we further fine-tuned a similar single-class detection model for fish, the MegaFishDetector [24], which has been trained using a combination of six underwater datasets.

In this paper, we compare strategies for using unlabeled data, in addition to a subset of labeled data, for biodiversity monitoring in two real-world underwater datasets collected by MBARI exhibiting long-tail distributions. We discuss the benefits of separating the localization and classification stages, training a single-class object detector for the localization step taking inspiration from a widely used and successful study on camera trap images. For the subsequent classification of the localized crops, we compare supervised, supervised iteratively, and semi-supervised approaches, showing that self-supervised and semi-supervised pre-training using unlabeled images followed by supervised fine-tuning results in a more balanced performance across all classes. For both datasets, contrastive methods that use a combination of unlabeled and labeled data for pretraining, such as PAWS and SimCLR with SuNCEt loss, resulted in significantly higher balanced accuracy scores in comparison to contrastive pretraining using unlabeled data alone, as is the case with standard SimCLR. On both datasets we achieved the highest balanced accuracy scores using semi-supervised pre-training using PAWS followed by supervised fine-tuning, however, the significant increase in balanced accuracy score comes at a cost of decreased overall accuracy. By proposing a pipeline consisting of a localization approach that is being widely used in practice, along with comparing classification approaches to make use of unlabeled data ranging from straightforward iterative supervision to newer methods such as semi-supervised pre-training, we anticipate that our results on two noisy real-world long-tailed datasets can serve as a guide for practitioners working on similar problems.

2. Methods

2.1. Datasets used

Two separate datasets were used for these experiments henceforth referred to as ROV (remotely operated vehicle) and AUV (autonomous underwater vehicle) datasets. The ROV dataset consists of 27,000 images collected from multiple ROV transects by MBARI. The dataset contains a mix of images taken in the benthic and midwater zones and was annotated by the video lab at MBARI with bounding box coordinates and label assignments (varying degrees of taxonomic assignment level). The ROV dataset contained 100 different animal labels. This resulted in a total of 41,000 localizations. The images were of varying resolution and consisted of animals of different sizes, ranging from small views of animals in the distance, to zoomed-in close-up shots of animals. The AUV consists of a high-resolution (2k) camera and moves at a speed of about 1 m/s underwa-

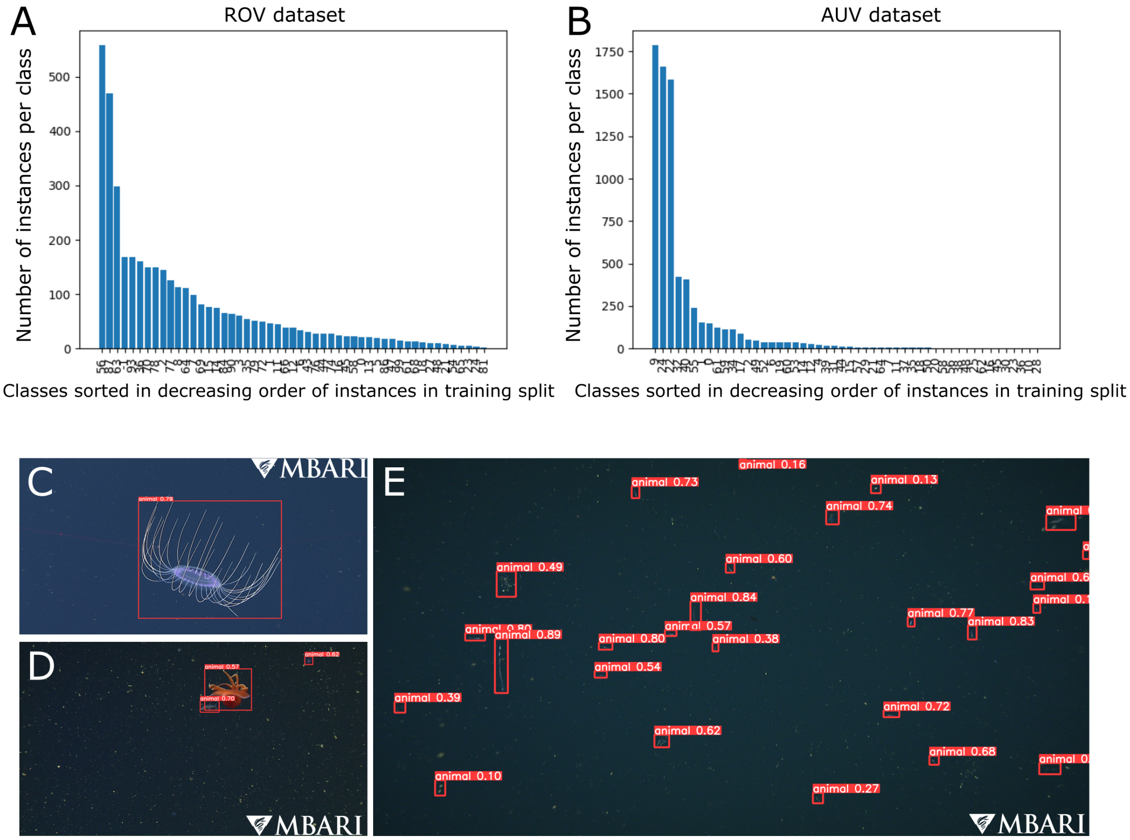

Figure 1. Long-tailed training data distribution and example images from two different underwater datasets collected by MBARI. (1A and 1B) Number of instances per class (class indices shown in place of taxonomic assignments for data embargo reasons) sorted from highest to lowest number of instances in the training splits of the ROV and AUV datasets respectively. These are the extracted crops from both datasets for classification. (1C and 1D) Example images from the ROV dataset with overlaid boxes for localization predictions with confidence scores from our trained single-class animal detection model. (1E) Example image from the AUV dataset with overlaid boxes for localization predictions with confidence scores from our trained single-class animal detection model.

Model	Test set	Precision	Recall	mAP50
Megafishdetector	ROV test split	0.541	0.445	0.39
Megafishdetector fine-tuned on ROV train split	ROV test split	0.74	0.74	0.783
Megafishdetector fine-tuned on ROV train split	AUV test split	0.719	0.512	0.647
Megafishdetector fine-tuned on ROV + AUV train split	AUV test split	0.689	0.687	0.739

Table 1. Comparison of localization performance of single-class animal detection models (YOLOv5 medium) on ROV and AUV test sets showing that initialization from Megafishdetector weights followed by fine-tuning on the respective training sets yields the best results.

ter. The AUV dataset consisted of 11,000 fully annotated images and 75000 localizations. The AUV dataset contained 70 different label assignments at different taxonomic levels. Annotations were done by Danelle E. Cline using a combination of manual labeling, heuristic methods such as blob detection and unsupervised methods such as clustering and manual verification. The larger field of view of the AUV camera, higher resolution, and higher speed of the vehicle, resulted in many more animal captures per frame, including those that are fast enough to escape the ROV. The large number of animals per frame along with the typically small size of localizations in this dataset made the annotation task very laborious. For both datasets only the first 50 animal labels, sorted from highest to lowest number of instances in the training split (discussed below), were retained and the remaining labels were grouped together as the ‘unknown’ class, resulting in 51 classes. This was done to study the effect of novel classes in the unlabeled data, as

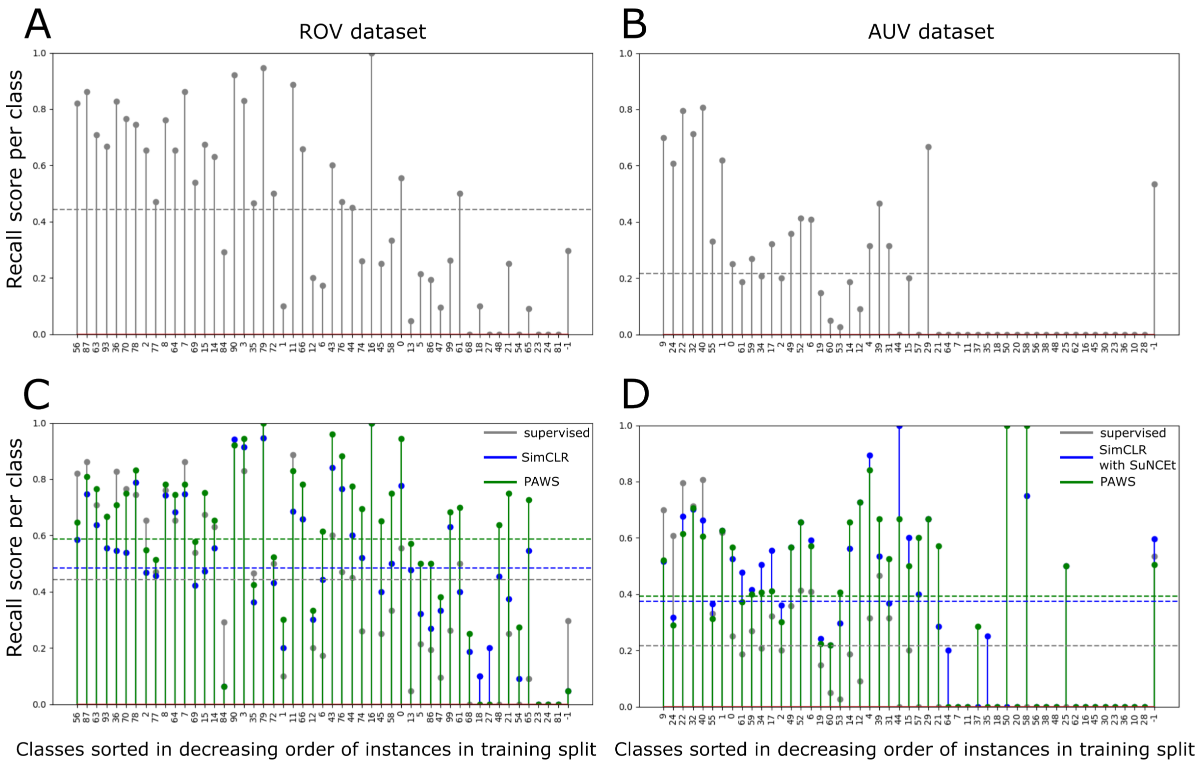

Figure 2. Comparison of supervised and semi-supervised approaches for classification on ROV and AUV datasets. (2A and 2B) Per-class recall scores on the ROV and AUV test splits respectively using supervised learning (fine-tuning on training splits starting from Imagenet weights). Classes are sorted in the same order (highest to lowest instances in the training set) as Fig. 1A,B except the unknown (-1) class which is at the end. The dotted line reflects the balanced accuracy score for each dataset. Supervised performance follows a long-tailed distribution. (2C) Per-class recall scores on the ROV test split comparing supervised, self-supervised (SimCLR) and semi-supervised (PAWS) pre-training on unlabeled split followed by supervised fine-tuning on the training split. (2D) Per-class recall scores on the test set of the AUV dataset comparing supervised, and two semi-supervised approaches (SimCLR with SuNCEt loss and PAWS) pre-training on a combination of labeled and unlabeled data followed by supervised fine-tuning on the training set.

Classification model	Overall accuracy	Balanced accuracy score
Supervised only: fine-tuning using training split + retaining labels of unlabeled split (max upper limit possible).	79.8	0.684
Supervised only: fine-tuning using training split starting from Imagenet.	64.97	0.443
Supervised iteratively - fine-tuning using training split + pseudo labels on unlabeled split thresholded at 0.1 confidence.	64.40	0.483
Supervised iteratively: fine-tuning using training split + pseudo labels on unlabeled split thresholded at 0.7 confidence.	**68.78**	0.472
SimCLR: Contrastive pre-training with NTXent loss on unlabeled split with batch size 256 for 100 epochs, followed by supervised fine-tuning using training split.	56.67	0.485
SimCLR: Contrastive pre-training with NTXent loss on unlabeled split with batch size 1028 for 100 epochs followed by supervised fine-tuning using training split.	53.03	0.473
PAWS: Contrastive pre-training on unlabeled split with unsupervised batch size 64 for 200 epochs followed by supervised fine-tuning using training split.	66.04	**0.587**

Table 2. Comparison of supervised, supervised iteratively, self, and semi-supervised classification performance on the ROV test set.

Classification model	Overall accuracy	Balanced accuracy score
Supervised only: fine-tuning using training split starting from Imagenet.	**62.74**	0.217
SimCLR: Contrastive pre-training with SuNCEt loss on unlabeled split with unsupervised batch size 64 for 100 epochs, followed by supervised fine-tuning using training split.	52.56	0.375
PAWS: Contrastive pre-training on unlabeled split with unsupervised batch size 64 for 200 epochs followed by supervised fine-tuning using training split.	49.58	**0.393**

Table 3. Comparison of supervised and semi-supervised classification performance on the AUV test set.

part of future work.

2.2. Dataset splits and unlabeled data

To simulate the availability of additional unlabeled data, annotations (bounding box coordinates plus assigned labels) from 75% of each dataset were removed and these images were treated as the unlabeled split. The remaining 25% of each dataset was split into train-val-test splits as 10-5-10%. All training was done on the train splits of each dataset and metrics were reported on their respective test split.

2.3. Evaluation metrics

For evaluating object detection, the standard metric of mAP50 was used. For evaluating classification performance, overall accuracy (OA) and balanced accuracy score (BA) (from sklearn [1]) were used. As both datasets were unbalanced and followed a long-tailed distribution, the overall accuracy may be skewed by the dominance of the majority class. Balanced accuracy score, the macro average of recall per class ranging from 0 to 1, is a better metric giving equal importance to performance on all classes irrespective of the number of instances per class.

2.4. Object detection

We trained a YOLOv5 [15] object detector model to predict bounding boxes in a class-agnostic manner by collapsing all classes into a general 'animal' class. We initialized this model with YOLOv5 parameters from a previously generalized fish detector called MegaFishDetector [24], fine-tuned on the training splits, and evaluated on respective test splits of both datasets. Once training and evaluation were complete, we were able to use the final model to extract crops of animals for the downstream classification task. Crops were extracted from images in the unlabeled splits only of both datasets as we already had annotated bounding box coordinates for train, val, and test splits. We evaluated the generalized object detector on its ability to correctly localize animals as measured by the mAP50 metric. A YOLOv5 medium model was used with the long edge of the image being 1280 pixels. A confidence threshold of 0.2 and an IoU threshold of 0.1 was used for predictions.

2.5. Classification

The localizations from our animal detector were cropped out, resulting in 31000 and 56000 extracted crops for the unlabeled splits in the case of the ROV and AUV datasets respectively. Annotated bounding box coordinates were cropped out from the train, val, and test splits of both datasets resulting in 4000, 2000, 4000 and 7500, 3750, 7500 extracted crops for the train, val, and test sets of ROV and AUV datasets respectively. Crops were resized to a size of 224 x 224 and fed into different classification models. For a fair comparison of the advantages of incorporating unlabeled data, baring small modifications (see section 4.3), the same model architecture, Resnet50 [12] trained for 200 epochs with a batch size of 128 using weighted cross-entropy loss was used for all supervised components of the different approaches. Models were initialized either from Imagenet weights or weights after contrastive pre-training on unlabeled splits.

2.5.1 Supervised (labeled data only). The baseline for comparison is training a Resnet50 following the standard supervised learning approach on the training split for each dataset. Models were initialized using Imagenet weights and were trained for 200 epochs using weighted cross-entropy loss, with weighting based on the number of instances of each class in the training split. The loss was monitored on the training and validation splits. Final evaluation metrics were calculated on the test split for each dataset.

2.5.2 Supervised iteratively. An approach that is not commonly compared to in semi-supervised learning papers, is the simple approach of using a trained model (supervised model from 2.5.1), to generate predictions on the unlabeled data, thresholding these predictions based on confidence, treating them as pseudo labels, and subsequently training a new model with a combination of the known labels plus pseudo labels on training split + unlabeled split that has been assigned a pseudo label (depends on threshold chosen). This iterative supervised approach, although simple, is a fair comparison for approaches making use of additional unlabeled data. Whereas ideally a confidence threshold for assigning pseudo labels would be picked based on a precision-recall curve, in this limited study we compared thresholds at two ends, 0.1 and 0.7.

2.5.3 Self-supervised pre-training: SimCLR. The first approach explored for incorporating additional unlabeled data was a contrastive learning approach, SimCLR. SimCLR has been shown to benefit from large models and large batch sizes. We compared batch sizes 256 and 1024, using a Resnet50 backbone with a prediction head (original model using in the SimCLR paper). Contrastive pre-training using unlabeled data was followed by supervised fine-tuning using the labeled data. The pre-training using contrastive NT-XENT loss was done for 100 epochs on unlabeled crops only using 4 GPUs (a ml.p3.8xlarge AWS instance) in the case of batch size 256 and 8 GPUs (a ml.p3.16xlarge AWS instance) in the case of batch size 1028. We only trained this model for 100 epochs as compared to 200 epochs due to the high financial cost associated with high GPU memory demands originating from the requirement of this method to have a large batch size. Augmentations used for pre-training were the same as in the original paper, random crop and color distortion. Pre-trained weights were then used as an initialization for supervised fine-tuning on the training split, for the same number of epochs (200) as in the supervised case.

2.5.4 Semi-supervised pre-training: PAWS and SimCLR with SuNCEt loss. The inclusion of a subset of labeled data during the contrastive pre-training step has been shown to result in faster convergence without the need to have a very large batch size, hence allowing cheaper GPU instances to be used. We explored two approaches that fall in this category, PAWS and SimCLR with SuNCEt loss. For both approaches, we used a relatively small unsupervised batch size of 64 and pre-trained for 200 epochs using 3 GPUs (a ml.g4dn.12xlarge AWS instance; there was a weird bug when trying to use 4 GPUs). Pre-trained weights were then used as an initialization for supervised fine-tuning on the training split, for the same number of epochs (200) as in the supervised case. PAWS is based on assigning soft pseudo-labels to unlabeled images based on their distances in feature space from a support set of labeled examples per class. The approach minimizes the cross-entropy loss between pseudo-labels assigned to two transformed versions of the same image. The support set of labeled examples is only used for pseudo-label assignment in the pre-training step. We tested the PAWS approach on both the ROV and AUV datasets. We tested SuNCEt loss, a semi-supervised loss that combines the SimCLR contrastive loss with an additional term aiming to distinguish labeled examples of different classes, only on the AUV dataset.

3. Results

3.1. Dataset distribution

Fig. 1A,B plot the number of images per class in sorted order from highest to lowest for the ROV and AUV train splits respectively. The actual class names (taxonomic assignments at various levels) are omitted for data embargo reasons. The top 50 classes in either dataset were retained and the rest were clubbed into a collective “unknown” class with an assigned index of -1 resulting in a total of 51 classes per dataset. Both datasets, like most datasets collected in the wild, exhibit a long tail distribution with many instances of common classes and some rare classes consisting of 2 or 3 images only. Although not shown, the validation and test sets also exhibit long-tail distributions.

3.2. Detection results

Tab. 1 shows the mAP50 scores of single-class (animal) detection starting from MegaFishDetector weights found online. Training on even a subset of data within the distribution of either dataset greatly increases performance. This is not surprising as deep networks struggle with out-of-distribution data. We use our final model to extract crops from unlabeled images in either dataset. Fig. 1C,D and Fig. 1E show examples of predicted bounding boxes on images in the ROV and AUV dataset test splits respectively.

3.3. Classification results

Once crops of animals are extracted either using our generalized animal detector in the case of unlabeled splits, or annotated coordinates in cases of the train, val, and test splits, they are resized to 224 x 224 and fed into a classification model to assign to one of the 51 classes. We compared supervised, supervised iteratively, and self and semi-supervised classification approaches incorporating additional unlabeled data for the two datasets. Tab. 2 and Tab. 3 present a comprehensive comparison of various classification approaches, encompassing both supervised and semi-supervised learning methods applied to the ROV and AUV datasets respectively.

3.3.1 Supervised only. To ascertain the upper limit of classification performance for the ROV dataset as a reference, we conducted an experiment retaining labels for the unlabeled split. We fine-tuned a supervised Resnet50 starting from Imagenet weights on a combination of the training split (10% of the dataset) plus the unlabeled split (75% of the dataset) and obtained a balanced accuracy score of 0.684. For the actual comparison in the limited labeled data regime, we fine-tuned supervised Resnet50 models on the ROV and AUV training splits each initialized from Imagenet weights. Fig. 2A,B show the per-class recall scores on the ROV and AUV test splits respectively. The class indices are sorted in the same order as Fig. 1A,B, i.e from highest to lowest number of instances in the training split. It is not surprising to see that the per-class performance also follows a long tail, as we know that deep networks perform poorly given a lower number of training examples. The dotted line shows the balanced accuracy score in either case, scores of 0.443 and 0.217.

3.3.2 Supervised iteratively. This assessment was exclusively conducted for the ROV dataset. To ensure a fair comparison with semi-supervised methodologies utilizing additional unlabeled data, we leveraged the trained supervised model (on the 10% training split only) from section 3.3.1 to generate predictions on images in the unlabeled split. Predictions were subjected to a thresholding process based on confidence, with predictions surpassing the threshold considered pseudo labels. While the optimal threshold selection typically involves a meticulous precision-recall curve analysis on the val set, we pragmatically assessed only two thresholds—0.7 and 0.1—for the sake of expediency. As shown in Tab. 2, this iterative supervised approach exhibits a modest enhancement in performance when increasing balanced accuracy score from 0.443 to 0.483 or 0.472 depending on the threshold. However, it is imperative to acknowledge the inherent risk of perpetuating biases learned during the initial supervised stage. Furthermore, any bias associated with the long-tailed nature of the dataset will be further emphasized, as only predictions with confidence exceeding the chosen threshold, usually the head classes, will contribute to additional pseudo labels.

3.3.3 Self-supervised pre-training: SimCLR. This assessment was exclusively conducted for the ROV dataset. For self-supervised contrastive pre-training approaches, we tested SimCLR using the original NT-Xent loss function. Contrastive pretraining was performed using the unlabeled split of the ROV dataset followed by supervised fine-tuning on the ROV training split and evaluation using the ROV test split. As we can see from Tab. 2 and Fig. 1C, this approach yielded only a modest improvement on the balanced accuracy score in comparison to the supervised only approach, improving balanced accuracy score from 0.443 to 0.485 while leading to a decrease in overall accuracy from 64.97 to 56.67 when using a batch size of 256. Increasing the batch size from 256 to 1024 did not result in significant gains. Pre-training was done for 100 epochs as opposed to 200 because of the limited improvements going from batch size 256 to 1024, along with the high financial cost associated with GPU memory requirements for this method that requires large batch sizes.

3.3.4 Semi-supervised pre-training: PAWS and SimCLR with SuNCEt loss. To test semi-supervised contrastive learning methods, we compared two approaches, SimCLR using SuNCEt loss, and PAWS on both the ROV and AUV datasets. These approaches use a combination of unlabeled data and a subset of labeled data for the pre-training step. Pre-training was followed by supervised fine-tuning as in 3.3.3. As we can see from Tab. 2 and Tab. 3, performing semi-supervised contrastive pre-training on the unlabeled split, followed by supervised fine-tuning on the training split results in a significantly higher balanced accuracy score sometimes at a cost of overall accuracy. This is also evident from the per-class performance of these models in Fig. 2C,D. We see a much more balanced performance, higher performance on rare classes plus slightly lower or the same performance on head classes. In the case of the ROV dataset, PAWS resulted in a significantly higher balanced accuracy score of 0.587 compared to standard SimCLR and supervised only methods yielding balanced accuracy scores of 0.485 and 0.443 respectively. In the case of the AUV dataset, we observe a similar significant gain in balanced accuracy score, nearly doubling the balanced accuracy score of supervised only approaches from 0.217 to 0.375 and 0.393 for SimCLR with SuNCEt loss and PAWS respectively, emphasizing the efficacy of these methods in handling dataset imbalances. In summary, for both the ROV and AUV long-tailed datasets, PAWS pre-training followed by supervised fine-tuning resulted in the highest balanced accuracy scores sometimes at the cost of overall accuracy (AUV dataset only and not ROV).

4. Discussion

We demonstrate that in the case of image classification in underwater datasets consisting of a subset of labeled data and a large amount of unlabeled data, semi-supervised pre-training methods such as SimCLR with SuNCEt loss and PAWS, followed by supervised fine-tuning using the labeled data, results in a significantly higher balanced performance across classes (Fig. 2C,D and Tab. 2, Tab. 3) when compared to supervised only baselines. This is especially apparent in cases of real-world datasets exhibiting a long-tailed distribution (Fig. 1A,B) as is most often the case with datasets collected in the wild. We make the case that splitting the localization and classification steps allows for training a robust generalized single-class detector (Fig. 1C-E and Tab. 1) which also helps address the open-world problem for localization. This subsequently allows training a suite of different classifiers depending on the task at hand, either supervised only for the best results on common classes, semi-supervised for the most balanced performance, classifiers focusing on few-shot learning for rare classes or classifiers addressing the open-world problem for classification.

4.1. Advantages of separating detection and classification stages

The deliberate separation of the localization step from the classification step presents several advantages compared to the conventional integration of these stages in standard object detectors, whether single-stage or two-stage. Training a single-class detector allows for the amalgamation of training data from diverse datasets by consolidating labels into a singular 'animal' class. This approach substantially enhances the model's generalizability and robustness. The widespread adoption of Megadetector, a generalized animal detector for camera-trap data on land, underscores the efficacy of this methodology. Beyond facilitating the integration of multiple datasets, this approach proves advantageous in the context of open-world detection. In scenarios involving previously unidentified species, a plausible occurrence in deep ocean exploration, our generalized detection model exhibits a higher likelihood of localizing the animal, having encountered diverse animal types from different backgrounds. Subsequently, the classification model can address the open-world scenario for classification, employing anomaly detection methods. In contrast, standard multi-class object detectors may entirely miss the animal due to a lack of resemblance to a limited training set of animal classes. Unlike self-supervised object detection approaches, which can be computationally intensive and offer marginal improvements over standard object detectors [14], the segregation of localization and classification steps not only capitalizes on the robustness of a single-class detector but also enables the exploration of self-supervised and semi-supervised learning strategies for utilizing unlabeled data for classification. These approaches are typically less computationally demanding and have demonstrated promising results. An additional benefit arising from the use of a single-class detector is the potential improvement in downstream tasks such as tracking, attributed to the absence of label switches from a multi-class object detector.

4.2. A more robust and balanced performance from pre-training

The utilization of unlabeled data for pre-training exposes models to the specific imaging domains they are intended to be trained on, enabling the acquisition of general features unique to marine imaging and marine animals. Incorporating random crop augmentation further facilitates the association of disparate segments of animals, even those that may exhibit gelatinous and structureless characteristics. In contrast to learning exclusively with labeled data, as observed in supervised cases, which compels the model to focus on features for maximal class distinction, incorporating unlabeled data is more likely to foster the learning of more general and robust features. Notably, prior studies have demonstrated that off-the-shelf semi-supervised models exhibit enhanced robustness to class imbalance compared to their fully supervised counterparts [17]. These models also demonstrate improved performance in out-of-distribution scenarios, cross-task settings, and rare class identification, and exhibit a balanced feature space equidistant from all classes and not dominated by the majority class as in supervised learning [16]. Our findings, based on two real-world underwater imaging datasets characterized by long-tail distributions, align with these observations. Notably, our results indicate a doubling of balanced accuracy in the case of the AUV dataset, which, despite the relatively small size and the blob-like appearance of individual animals due to their distance from the vehicle, underscores the efficacy of our approach. Compared to supervised learning which can perpetuate biases, our approach yields more balanced results and is particularly beneficial when data are limited.

4.3. Model architectures differ slightly

As detailed in the methods section, it is crucial to note that the original models employed in the SimCLR and PAWS studies, as well as the models utilized for our semi-supervised pre-training, deviate from the standard Resnet50 configuration. Specifically, they feature a Resnet50 architecture augmented with an additional prediction head. While the ideal comparison involves assessing identical architectures across both supervised and semi-supervised approaches, it is improbable that the observed improvement in balanced accuracy scores can be solely attributed to the presence of the supplementary feedforward prediction layer in these models. The exploration of a direct comparison using identical architectures is an ongoing aspect of our research.

4.4. Semi-supervised pre-training works better than self-supervised pre-training for imbalanced datasets

From section 3.3.4, it is clear that semi-supervised pretraining approaches that use a combination of unlabeled and labeled data for pretraining, such as SimCLR with SuNCEt loss and PAWS, result in significantly higher balanced accuracy scores on the long-tailed distributed test sets for both the ROV and AUV datasets. These approaches also require significantly lower compute cost and time in comparison to self-supervised pretraining approaches like SimCLR. One can see how providing some supervisory signal by using a subset of labeled data, can result in faster convergence. We have shown that the weights converged onto by using the additional supervisory signal, result in a greater robustness to dataset imbalance, leading to significantly higher balanced accuracy scores after supervised fine-tuning in comparison to self-supervised pre-training approaches, supervised only and supervised iteratively on two real-world long-tailed underwater datasets.

References

[1] sklearn.metrics.balanced_accuracy_score — scikit-learn 1.4.1 documentation, . 5

[2] Who is using Megadetector? - https://github.com/agentmorris/megadetector?tab=readme-ov-file#who-is-using-megadetector, . 2

[3] Mahmoud Assran, Nicolas Ballas, Lluis Castrejon, and Michael Rabbat. Supervision Accelerates Pre-training in Contrastive Semi-Supervised Learning of Visual Representations, 2020. arXiv:2006.10803 [cs, stat]. 2

[4] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Armand Joulin, Nicolas Ballas, and Michael Rabbat. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8443–8452, 2021. 2

[5] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, and Yuandong Tian. A cookbook of self-supervised learning. *arXiv preprint arXiv:2304.12210*, 2023. 1

[6] Sara Beery, Dan Morris, and Siyu Yang. Efficient Pipeline for Camera Trap Image Review, 2019. arXiv:1907.06772 [cs]. 2

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2

[8] Elijah Cole, Xuan Yang, Kimberly Wilber, Oisin Mac Aodha, and Serge Belongie. When does contrastive visual representation learning work? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14755–14764, 2022. 1

[9] Ellen M. Ditria, Sebastian Lopez-Marcano, Michael Sievers, Eric L. Jinks, Christopher J. Brown, and Rod M. Connolly. Automating the Analysis of Fish Abundance Using Object Detection: Optimizing Animal Ecology With Deep Learning. *Frontiers in Marine Science*, 7:429, 2020. 1

[10] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *Proceedings of the ieee/cvf International Conference on computer vision*, pages 6391–6400, 2019. 1

[11] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, and Armand Joulin. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021. 1

[12] 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. 5

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020. 1

[14] Gabriel Huang, Issam Laradji, David Vazquez, Simon Lacoste-Julien, and Pau Rodriguez. A survey of self-supervised and few-shot object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4071–4089, 2022. Publisher: IEEE. 2, 8

[15] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, Kalen Michael, TaoXie, Jiacong Fang, Imyhxy, Lorna, ”Zeng Yifu” , Colin Wong, Abhiram V, Diego Montes, Zhiqiang Wang, Cristi Fati, Jebastin Nadar, Laughing, UnglvKitDe, Victor Sonck, Tkianai, YxNONG, Piotr Skalski, Adam Hogan, Dhruv Nair, Max Strobel, and Mrinal Jain. ultralytics/yolov5: v7.0 - YOLOv5 SOTA Realtime Instance Segmentation. *Zenodo*, 2022. ADS Bibcode: 2022zndo...7347926J. 5

[16] Bingyi Kang, Yu Li, Sa Xie, Zehuan Yuan, and Jiashi Feng. Exploring balanced feature spaces for representation learning. In *International Conference on Learning Representations*, 2020. 1, 8

[17] Hong Liu, Jeff Z. HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised Learning is More Robust to Dataset Imbalance, 2022. arXiv:2110.05025 [cs, stat]. 1, 8

[18] Md Kislu Noman, Syed Mohammed Shamsul Islam, Jumana Abu-Khalaf, and Paul Lavery. Multi-species seagrass detection using semi-supervised learning. In *2021 36th International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pages 1–6. IEEE, 2021. 2

[19] Mehdi Noroozi and Paolo Favaro. Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. In *Computer Vision – ECCV 2016*, pages 69–84. Springer International Publishing, Cham, 2016. Series Title: Lecture Notes in Computer Science. 1

[20] Omiros Pantazis, Gabriel J. Brostow, Kate E. Jones, and Oisin Mac Aodha. Focus on the positives: Self-supervised learning for biodiversity monitoring. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10583–10592, 2021. 2

[21] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016. 1

[22] Penny Tarling, Mauricio Cantor, Albert Clapés, and Sergio Escalera. Deep learning with self-supervision and uncertainty regularization to count fish in underwater images. *PloS one*, 17(5):e0267759, 2022. Publisher: Public Library of Science San Francisco, CA USA. 2

[23] Devis Tuia, Benjamin Kellenberger, Sara Beery, Blair R. Costelloe, Silvia Zuffi, Benjamin Risse, Alexander Mathis, Mackenzie W. Mathis, Frank van Langevelde, Tilo Burghardt, Roland Kays, Holger Klinck, Martin Wikelski, Iain D. Couzin, Grant van Horn, Margaret C. Crofoot, Charles V. Stewart, and Tanya Berger-Wolf. Seeing biodiversity: perspectives in machine learning for wildlife conservation. *Nature Communications*, 13(1):792, 2022. arXiv:2110.12951 [cs]. 1

[24] Daniel Yang, Levi Cai, Stewart Jamieson, and Yogesh Girdhar. Biological Hotspot Mapping in Coral Reefs with

Robotic Visual Surveys. *arXiv preprint arXiv:2305.02330*, 2023. 2, 5

[25] Richard Zhang, Phillip Isola, and Alexei A. Efros. Colorful Image Colorization. In *Computer Vision – ECCV 2016*, pages 649–666. Springer International Publishing, Cham, 2016. Series Title: Lecture Notes in Computer Science. 1