


Beyond transfer learning: Leveraging ancillary images in automated classification of plankton

Jeffrey S. Ellen ^{1,2*} Mark D. Ohman ²

¹Basic and Applied Research Division, NIWC Pacific, San Diego, California, USA

²California Current Ecosystem LTER site, Scripps Institution of Oceanography, University of California, San Diego, La Jolla, California, USA

Abstract

We assess whether a supervised machine learning algorithm, specifically a convolutional neural network (CNN), achieves higher accuracy on planktonic image classification when including non-plankton and ancillary plankton during the training procedure. We focus on the case of optimizing the CNN for a single planktonic image source, while considering ancillary images to be plankton images from other instruments. We conducted two sets of experiments with three different types of plankton images (from a *Zooglider*, Underwater Vision Profiler 5, and Zooscan), and our results held across all three image types. First, we considered whether single-stage transfer learning using non-plankton images was beneficial. For this assessment, we used ImageNet images and the 2015 ImageNet contest-winning model, ResNet-152. We found increased accuracy using a ResNet-152 model pretrained on ImageNet, provided the entire network was retrained rather than retraining only the fully connected layers. Next, we combined all three plankton image types into a single dataset with 3.3 million images (despite their differences in contrast, resolution, and pixel pitch) and conducted a multistage transfer learning assessment. We executed a transfer learning stage from ImageNet to the merged ancillary plankton dataset, then a second transfer learning stage from that merged plankton model to a single instrument dataset. We found that multistage transfer learning resulted in additional accuracy gains. These results should have generality for other image classification tasks.

Of the types of algorithms that can be employed in image analysis, “supervised classification” is a term used to describe machine learning algorithms that assign one or more labels to the image from a predetermined list of labels. This approach is usually based on specific “regions of interest” (ROIs) that are labeled to facilitate further analysis, such as population density estimates. Supervised classification algorithms require example images with labels already assigned, commonly referred to as “training images.” Given additional unlabeled images as input, the algorithm calculates which group of labeled images is the most similar to the current image and then provides that most similar label as output. In general, the more labeled images that are provided as training data, the higher the resulting accuracy, although various assessments

have quantified that there are diminishing returns on training set size. For example, Sun et al. (2017) found that “performance on vision tasks increases logarithmically based on volume of training data size.”

Manually labeling images is labor intensive. An alternative to using only one’s own data is to leverage transfer learning, where a machine learning model trained on other images is reused to initiate training on the images of interest. Transfer learning is not a new concept; Thrun and Pratt (1998) discussed the theory and application of transfer learning, including a survey of work done up to that point. In one of the earliest and most widely cited applications of transfer learning with CNNs, Girshick et al. (2014) found a large gain in accuracy when conducting “supervised pretraining for an auxiliary task, followed by domain-specific fine-tuning” compared to training on only the smaller domain-specific dataset by itself.

Here, we assess whether transfer learning training of a model improves classification accuracy with planktonic datasets relative to training de novo with only the native images. Although our design does not require the images to be novel, we believe our results are particularly useful when trying to create a model for the first time and/or the existing library of labeled plankton

*Correspondence: jeffrey.s.ellen.civ@us.navy.mil, jellen@ucsd.edu

Additional Supporting Information may be found in the online version of this article.

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial](https://creativecommons.org/licenses/by-nc/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

images is small. This situation commonly arises when newly working with an imaging device or with deployments in a new environment.

We also describe and assess a technique incorporating additional planktonic image datasets through sequential rounds of tuning, starting with ImageNet, tuning on ancillary plankton images, and tuning again on only the target images. A similar approach has been described at least three times in recent literature, all with inconclusive results. Lumini et al. (2023) built ensembles of CNNs, and they include as constituents individual CNNs trained using “two rounds tuning (2R).” Lumini et al. used five small oceanographic datasets (three plankton datasets of 3771; 6600; and 14,374 images, and three coral datasets of 766 and 1123 image patches). After starting with ImageNet, individual coral and plankton datasets were used to train other individual coral and plankton datasets, respectively. They found minimal benefit in 2R individually. Guo et al. (2021) also used five datasets, four non-plankton datasets (flowers, seedlings, and fish) ranging from 1360 to 8189 images and a planktonic dataset of 60,736 images. In their “multistage transfer learning,” they comprehensively compared every permutation of multistage transfer learning on four different network sizes, and found minimal to no gains in most cases (they did find that ImageNet provided the best results, and the next largest library, the planktonic one,

always provided the next best starting point for transfer learning to every other domain). Orenstein and Beijbom (2017) performed “double fine-tuning” on two planktonic datasets of 60k and 120k images. They found accuracy gains of less than one percentage point (Orenstein and Beijbom 2017).

Our technique goes beyond these previous approaches in two respects. First, we use larger datasets. Second, we combine our ancillary images into a single, much larger and more complex dataset containing images from multiple acquisition systems. We find that utilizing all ancillary images simultaneously increases accuracy. Our motivation for including ancillary plankton images is to not only improve algorithm performance on the supervised classification task, but also to leverage human expertise from multiple laboratories and a wide variety of taxonomic specializations. Our results indicate that this technique is effective for ML models on multiple types of plankton images, and likely for other types of images as well.

Materials and procedures

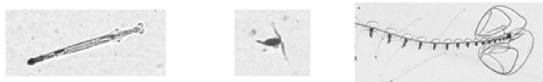
Datasets

We used three digital image datasets in our experiments: *Zooglider* images acquired in situ (Ohman et al. 2019), Zooscan images acquired in the laboratory (Gorsky et al. 2010), and in situ Underwater Vision Profiler 5 images (Picheral et al. 2010).

We also indirectly leveraged the ImageNet dataset. The ImageNet Large Scale Visual Recognition Challenge has been held annually since 2010 (Russakovsky et al. 2015). The contest was enabled through human annotation of millions of images obtained from the internet (Deng et al. 2009). We used a machine learning model trained on the version of the competition dataset used from 2012 to 2015, which comprised 1.28M images from 1000 different classes (Russakovsky et al. 2015; He et al. 2016). ImageNet has been used extensively in Computer Vision applications, and the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) awarded ImageNet “retrospective most impactful paper from CVPR 2009” (Martinez 2019). Some example images are shown in Fig. 1, assigned labels such as “bell pepper” and “pizza/pizza pie.” The granularity of the classes is at the level of a layperson (e.g., “fly,” “bee,” and “cricket” for insects, and “jellyfish,” “sea anemone,” and “sea urchin” for aquatic organisms). A single label is assigned to the object that is the largest and/or closest to the center of the image, although other recognizable objects may also be in the image (Fig. 1).

Zooglider images were acquired by a shadowgraph imaging Zoocam mounted on an autonomous *Zooglider* (Ohman et al. 2019). Images are captured during glider ascent between 400 m and the sea surface. Illumination is provided by a light-emitting diode centered at 620–630 nm, so as to minimize animal avoidance (Ohman et al. 2019). The imaged volume is a 250 mL cylinder of collimated light. The captured images are 1296 × 964 with a pixel resolution of 40 μm, and the system

- *Zooglider* (in situ, 58 classes, 1.2M images)



- ZooScan (preserved, 30 classes, 2.1M images)



- UVP5 (in situ, 42 classes, 145k images)



- ImageNet (non-plankton, 1000 classes, 1.28M images)



Fig. 1. Example images from each of the four datasets used (*Zooglider*, Zooscan, UVP5, and ImageNet), with an indication of the site of imaging (in situ or preserved for zooplankton images), the number of classes into which objects were classified, and the total number of annotated images used. For the three zooplankton sources, similar taxa are arranged in columns. For ImageNet, four images were selected at random.

is optimized for mesozooplankton ranging in size from approximately 0.5–30 mm (Ohman et al. 2019). A segmentation algorithm (Ellen et al. 2019) is applied to identify “regions of interest” (ROI) within each image, so as to have only a single plankter in each ROI. We developed a training set of 1,211,653 ROI with manual annotations corresponding to one of 58 classes (Supporting Information Table S1). *Zooglider* ROI were acquired in California Current System Long Term Ecological Research (CCE-LTER) waters across 1511 dives from 14 different multiday deployments spanning 2017–2020.

ZooScan is a commercially available instrument used to create digital images of preserved plankton (Gorsky et al. 2010). Unlike the other two image sets, Zooscanned zooplankton were collected with a net and preserved, both steps that can result in changes in tissue opacity, altered postures, shrinkage, or missing appendages. ZooScan images are captured as a single intensity channel with gray level normalization and pixel resolution of 10.6 μm (Gorsky et al. 2010). The ZooScan hardware captures the entire imaging field in a single pass, and then ImageJ-based software executes segmentation such that every contiguous area of dark pixels is saved as a rectangular ROI. ROIs are augmented with small black lines in two of the four corners of the bounding rectangle, and the ROI saved has a margin of a few pixels beyond the bounding rectangle. A scalebar is added in the bottom margin. All of these annotations were removed prior to machine learning experiments. For these experiments, we developed a training set of 2,138,292 manually annotated Zooscan images corresponding to 1 of 30 classes (Supporting Information Table S1). The images were obtained from plankton samples collected in the CCE-LTER region on CalCOFI cruises between May 2007 and July 2020.

Underwater Vision Profiler (UVP5; Picheral et al. 2010) images were acquired between 2008 and 2019 as part of the CCE-LTER process cruise studies (<https://ccelter.ucsd.edu/cruise-documents/>). The UVP5 was lowered on a CTD-rosette at 30–60 m min^{-1} . UVP5 images were acquired in situ with a fixed focal lens aimed perpendicular to a sheet of water illuminated by a collimated 625 nm LED (Picheral et al. 2010). Each frame consists of a 22×18 cm volume imaged as 1280×1024 pixels (pixel resolution 174 μm). For these experiments, we developed a training set of 145,419 annotated images assigned to one of 42 classes (Supporting Information Table S1).

For all three plankton image libraries used in our experiments, roughly 40% of the images are detritus, artifacts, or unknown, 40% of the images are copepods, and 20% are other biological objects.

We aggregated these 3.45 million images in two ways. First, we concatenated the datasets, resulting in 130 classes, each consisting only of images from a single instrument (58 *Zooglider* + 30 ZooScan + 42 UVP5), which we refer to as the “combined” dataset. Second, we assemble a dataset based on aligning the same image categories from different instruments. This procedure resulted in a new set of labels consisting of 50 classes

for these 3.45 million images, most of which consisted of images from multiple instruments, which we refer to as the “aligned” dataset. Note that the construction of the “combined” dataset requires reduced plankton taxonomic knowledge, but the construction of the “aligned” dataset requires considerable expertise to judge whether or not classes from different instruments should be considered equivalent. As constructed, each of these datasets, therefore, contains 40% ancillary images for *Zooglider*, 65% ancillary images for ZooScan, and 95% ancillary images for UVP5.

Computing hardware

Most experiment replicates were executed on a server with multiple Tesla V100 SXM2 GPUs, each of which has 32GB of RAM. Alternatively, a server with a single Tesla K40c with 12GB of RAM was used for early trials. Depending on the specific configuration and dataset, most models occupied 10–12GB of GPU memory while executing. GPUs were configured to use CUDA 11.4. Convolutional neural network models were built using PyTorch (Paszke et al. 2019), specifically torch 1.8.0 + cu111, torchvision 0.9.0 + cu111. Underlying notable dependencies used were Python 3.7.4, conda 4.12.0, and Numpy 1.17.2.

CNN architectures used

We selected ResNet-152 (He et al. 2016) as the basis for our initial transfer learning experiments because of its wide usage. ResNet-152 won the ImageNet classification contest in 2015 (Russakovsky et al. 2015). PyTorch provides the ResNet network structure available as well as a set of weights resulting from training on ImageNet data (Paszke et al. 2019). We conducted our experiments using 152 layers because it was the largest pretrained network available, thus providing the best results, with the tradeoff of requiring additional computational resources and training time.

ResNet architectures feature two notable implementation decisions. First is that ResNets include a ReLU, or Rectified Linear Unit, activation function that is applied to the output of the convolutional layers (Supporting Information Fig. S1a). ReLU activation is used in place of sigmoid activation because very large activations still have a gradient. Sigmoid activations, however, have gradients that approach zero. Another advantage of ReLU is that individual calculations are quicker because ReLU obviates the need for millions of exponential arithmetic computations during each epoch to fit each neuron’s output to a sigmoid curve. In addition, the ResNet architecture also includes shortcut connections, where a copy of the output from one layer is supplied as additional input to a later layer (Supporting Information Fig. S1a,b).

Procedures—Preprocessing

The pretrained ImageNet model requires three-channel (RGB) images of size 128×128 pixels as input. All three of our instruments acquire single-channel images, so we cloned

that channel to match the expected number of input channels rather than discard pretrained filters from ImageNet.

All images from the three datasets were variously sized rectangles with scalebars and accompanying text that needed to be removed. Our preprocessing, therefore, consisted of three steps (Fig. 2). First, we cropped each image to remove the scalebars. Next, we added empty pixels (padding) to either the horizontal or vertical sides of the image to create a square ROI.

Padding was split evenly to keep the ROI centered (e.g., an equal number of rows was added to the top and bottom of the image). The specific value of the pixel padding depended on the instrument; for the ZooScan and UVP5, we padded with pure white pixels to match the background, and for the *Zooglider* we padded with random pixel values, where the pixel values were selected from a Gaussian distribution centered on the average greyscale value of all images, and a variance that resulted in a speckled background consistent with most *Zooglider* images. Finally, we resized each ROI to 128×128 pixels. If the ROI was larger than 128×128 , we downscaled the image. If the ROI was smaller than 128×128 , we padded the image with more empty pixels to avoid introducing artifacts. For our datasets, roughly 95% of all ROIs were smaller than 256×256 . After finding in limited experiments that larger network sizes did not provide noticeable accuracy gains, we used the 128×128 ImageNet pretrained model.

Procedures—Training models

For each model, we split the dataset into 80% training data, 10% validation data, and 10% test data. We trained models to convergence, stopping after 10 consecutive epochs showed no reduction in validation loss (Fig. 3c). For each trial, we specified Python’s random seed, such that a specific

train/validation/test split was used for every applicable treatment, thereby preventing some treatments from getting “easier” splits. We conducted five replicates per treatment.

Our hyperparameter tuning was limited to the initial exploration conducted with the goal of finding values that performed adequately with all three data types. We selected hyperparameters that resulted in training and validation loss curves that iteratively trained a model as expected (Fig. 3c) and produced optimal accuracy on the training data. In Fig. 3a,b, we illustrate training and validation loss curves that the user should seek to avoid. Our search included an evaluation of the learning rate, three different dropout rates, and three different optimizers, primarily on ResNet-18 models, with some validation on ResNet-50 and ResNet-152. For ResNet-152 models, we did not tune hyperparameters individually per dataset or per trial, to avoid confounding factors in our transfer learning analysis. Accordingly, our overall accuracy should be treated as a floor. For these trials, we used a weight decay of 0.00005 and a learning rate of 0.00005.

When utilizing transfer learning, if the two image libraries have a different number of labeled classes, then the existing model cannot be used as is, and one or more layers need to be modified. For example, an architecture trained on ImageNet will produce 1000 class probabilities, and it is unlikely that the target plankton image library has exactly 1000 different labeled classes. At a minimum, the last fully connected layer needs to be modified to match the new label quantity. Since at least one layer needs to be altered, most practitioners choose to replace all fully connected layers (Fig. 4). In this manner, the convolutional layers can be thought of as “feature extractors” which generate input for the fully connected layers, which can be thought of as the “classifier.” The least computationally intensive approach to fit this model to new data is to

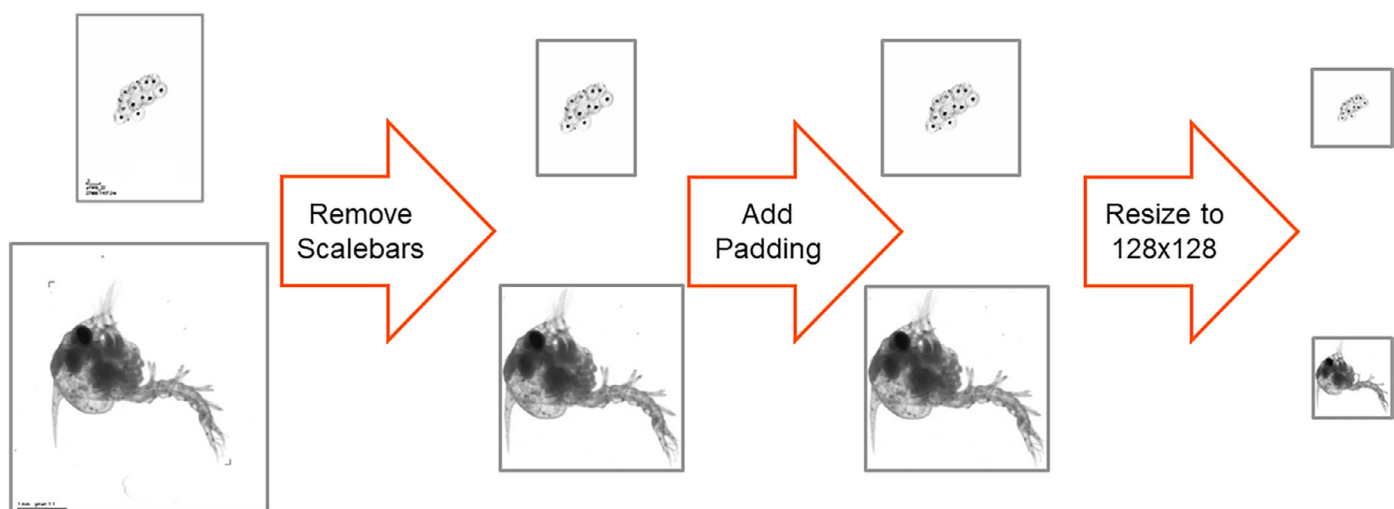


Fig. 2. Image preprocessing steps. First, cropping is executed to remove scalebars and text. Next, the image is padded (if needed) so that the height and width match (square grid of pixels). Next, each image is resized to a consistent 128×128 matrix.

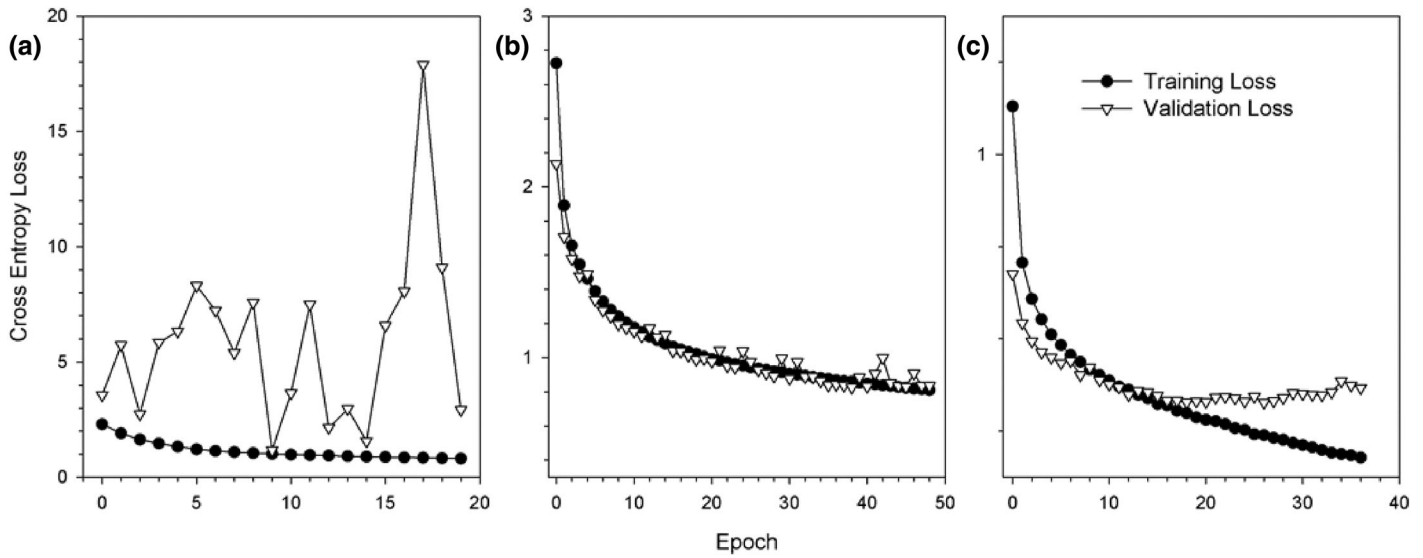


Fig. 3. Examples of **(a, b)** undesirable and **(c)** desired training and validation loss curves. **(a)** Erratic validation loss, which increases by an order of magnitude from its lowest point. **(b)** Results after reducing the learning rate by an order of magnitude. With the network weights being adjusted more gradually, the validation loss declines but shows no evidence of reaching a minimum. **(c)** Smooth, continuous training loss, which indicates the weights are being adjusted by an appropriate amount. Here, validation loss reaches a minimum at epoch 26. Since our stopping condition was 10 epochs without improvement to the validation set loss, training concluded after 36 epochs.

leave the convolutional layers alone and modify only the weights in these final classification layers (e.g., Mitra et al. 2019; Rodrigues et al. 2018). Mitra et al. (2019) modified and tuned the fully connected layer structure to classify one of seven different classes of foraminifera. Rodrigues et al. (2018) completely replaced the fully connected layers with a support vector machine (SVM) to serve as the classifier and then tuned the SVM to label their 20 classes of plankton. Note that in these cases, the image libraries totaled only 1437 and 5175 images, respectively. The more computationally intensive approach is to train all network layers (convolutional and fully connected) on the target images (e.g., Orenstein and Beijbom 2017). In our transfer learning experiments, we directly compared the efficacy of retraining only the modified fully connected layers with fine-tuning all network layers.

For our assessment of the efficacy of ancillary images, we executed transfer learning twice per experiment (Fig. 5). For the first iteration, we started with a pretrained ImageNet model, replaced the last fully connected layer of the network with one of the appropriate sizes for our number of target classes, and tuned all layers while training on either our “combined” or “aligned” dataset (Fig. 5a,b). After this initial transfer, we conducted a second round of transfer learning, again replacing the last layer of the network, then training to converge on a single instrument’s images (Fig. 5c,d). We used the same train validation test split for individual images from the target instrument type, so that the network was not getting an artificial boost from having images in the test set of the second round of

transfer learning that had been used as training images in the first round.

Performance metrics

We use unweighted Top-1 accuracy as our primary performance metric. Top-1 means that even though the model may predict non-zero probabilities for more than one class, only the highest probability label is used in the accuracy computation (i.e., no partial credit). Unweighted means that all labels are treated with equal weight, that is, the model does not disproportionately benefit from identifying any particular class over another. We report a number of epochs rather than time elapsed because we were running multiple trials in parallel, sometimes using the same GPUs, and always using the same shared system memory and hard drive on a shared system. Therefore, elapsed time varied depending on the system load during execution. An example confusion matrix illustrating class-specific accuracies may be seen in Supporting Information Fig. S2.

Assessment

Is transfer learning beneficial?

We first consider whether transfer learning results in more accurate classifications for a plankton image dataset when compared with a same-sized network initialized with random weights and trained only with the target plankton images. For the three plankton datasets considered (*Zooglider*, *ZooScan*, *UVP5*), a ResNet-152-sized model with random initialization trained de novo on plankton images provided

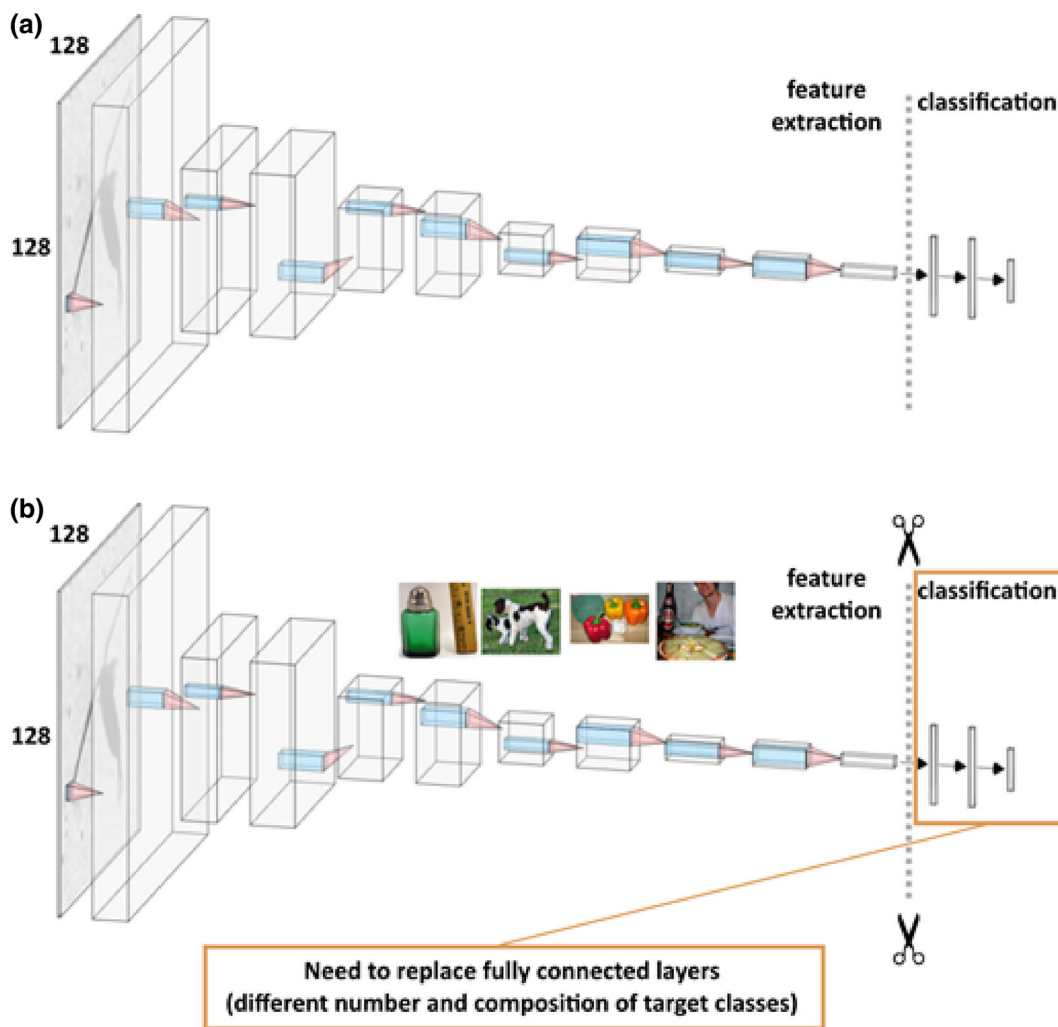


Fig. 4. (a) A convolutional neural network (CNN) with a 128×128 -pixel copepod image being used for computation against five convolutional/pooling layers (rectangular prisms) to the left of the dotted line, and three fully connected layers to the right of the dotted line. (b) A CNN in a transfer learning experiment, where all network layers are loaded using weights derived from training against a different dataset, such as ImageNet (shown here as household objects/animals) before the three fully connected layers are replaced with slightly different-sized layers (in orange outline).

overall mean accuracy ranging from 94.25% to 91.85% to 84.28% for the three datasets, respectively (Table 1, top row). Transfer learning from a ResNet-152 model pretrained using ImageNet images, and tuning only the fully connected layers on one of the types of plankton images (cf. Fig. 4), resulted in considerably lower classification accuracy and often a larger number of epochs (Table 1, second row). The best accuracy for all three types of plankton images was obtained when using transfer learning from a ResNet-152 model pretrained using ImageNet images, but tuning all weights in all layers of the network against the plankton images (Table 1). In addition to the highest accuracy, tuning all layers resulted in the smallest number of epochs required to obtain a stable solution. Therefore, we conclude that transfer learning is beneficial for both accuracy and computational resource usage for plankton image classification.

Is there additional accuracy gain resulting from transfer learning with ancillary images?

We then consider whether the use of ancillary images that originate from different types of imaging devices, together with multiple rounds of training, provides better results than transfer learning from ImageNet alone. For the three plankton datasets considered (*Zooglider*, *ZooScan*, *UVP5*), the extra transfer learning step using the Combined dataset resulted in overall mean accuracies of 95.14%, 92.87%, and 87.52%, equating to gains of 0.59, 0.57, and 2.09 percentage points, respectively (Table 2). Using the Aligned dataset for the extra transfer learning round of training provided gains over ImageNet alone, but less than the combined strategy, and also required more training epochs. The first step of training the network on the aligned dataset required an average of 47 epochs, while training the network on the combined

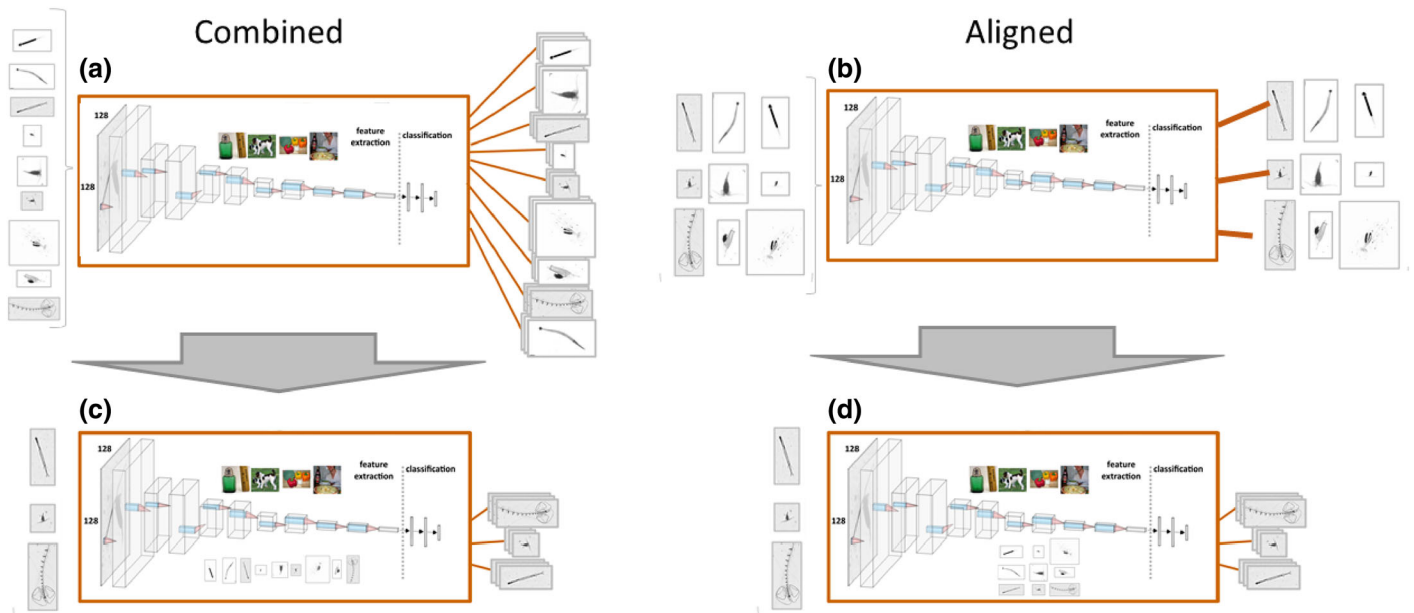


Fig. 5. Transfer learning from ImageNet on our (a, c) combined and (b, d) aligned datasets. (a) Combined: each class from each instrument is treated as its own class (e.g., *Zooglider* Chaetognath and *Zooscan* Chaetognath are assigned separate labels). (b) Aligned: each class is defined as a grouping of related classes from all instruments (e.g., *Zooglider* Chaetognath and *Zooscan* Chaetognath are assigned the same label). (c, d) Each is then used as the basis for a second transfer learning step using the image classes for a single instrument (here *Zooglider*).

Table 1. Training a CNN de novo on a ResNet-152 size model compared with transfer learning from a ResNet-152 model pretrained using ImageNet images. For each of the three imaging devices (*Zooglider*, *ZooScan*, and *UVP5*), the best results (as % accuracy of trained model [mean \pm 95%] and minimum number of epochs required) are obtained when performing transfer learning from ImageNet and tuning the weights of all layers.

Experiment	Zooglider		ZooScan		UVP5	
	Accuracy (%)	Epochs (N)	Accuracy (%)	Epochs (N)	Accuracy (%)	Epochs (N)
ResNet-152 de novo (random initialization)	94.25 \pm 0.05	68	91.85 \pm 0.41	51	84.28 \pm 0.42	102
Transfer from ImageNet (tuning fully connected layers only)	86.11 \pm 1.76	90	80.55 \pm 0.67	80	67.24 \pm 0.30	108
Transfer from ImageNet (tuning all layers)	94.55 \pm 0.06	29	92.30 \pm 0.06	31	85.43 \pm 0.25	35

Table 2. Transfer learning from ImageNet (from Table 1, tuning all layers) compared with multistage transfer learning using the ancillary images formulated as the aligned and combined datasets (see Fig. 4c,d). Both aligned and combined approaches provide gains (as % accuracy of trained model [mean \pm 95%]); however, the gains are greater using the combined network for all three cases.

Experiment	Zooglider		ZooScan		UVP5	
	Accuracy (%)	Epochs (N)	Accuracy (%)	Epochs (N)	Accuracy (%)	Epochs (N)
Transfer from ImageNet	94.55 \pm 0.06	29	92.30 \pm 0.06	26	85.45 \pm 0.34	37
Transfer from aligned	94.96 \pm 0.22	62	92.44 \pm 0.29	65	86.59 \pm 0.61	74
Transfer from combined	95.14 \pm 0.06	54	92.87 \pm 0.05	53	87.52 \pm 0.13	59

dataset required an average of 40 epochs. The number of epochs reported in Table 2 rows 2 and 3 is cumulative. For example, the transfer learning step starting with the 40-epoch combined model and training on *Zooglider* images

required an average of 14 epochs, resulting in a cumulative average of 54 epochs. Therefore, we conclude that the use of ancillary images is beneficial for accuracy, but aligning the datasets is not worth the extra effort.

Discussion

For all three of our plankton image datasets, we found that retraining the entire network provided better results than training only the fully connected layers and also converged in fewer epochs. One possible reason is that ImageNet consists of full-color, full-scene heterogeneous images, while the transfer target is millimeter-scale plankton images with more similar backgrounds. Our assumption is that when the convolutional layer weights are fixed, the ImageNet-only “feature extractor” produces a noisy signal regarding the components of the plankton image, which hampers the overall network performance. This noisy “feature extractor” then requires more epochs to converge than the cleaner signal produced by the convolutional layers when their weights are fine-tuned to be, more specifically, a “plankton extractor.” We find it notable that in all three cases, however, including ImageNet was advantageous.

While we only evaluated training all convolutional layers or one, convolutional layers could be trained or fixed in many permutations, such as fixing the first few layers and allowing the rest to be retrained. Yosinski et al. (2014) provide a good summary of the tradeoffs of each approach as well as a quantitative assessment of different combinations of retraining vs. fixing varying numbers of convolutional network layers. They found that splitting the ImageNet dataset in half (each half with 645,000 images in 500 classes), using the full network (i.e., copying all layers of weights), and then allowing all layers to be retrained yields the best result. They also found that transfer learning while freezing all of the convolutional layers is detrimental, yielding approximately 10% lower Top-1 accuracy than just using a randomized network initialization. Our results agree well with this finding.

Yosinski et al. (2014) specifically found that transfer learning from one half of ImageNet to the other provided a 2.1% boost in Top-1 accuracy over the baseline. Since their Top-1 baseline accuracy was around 62%, this boost is approximately a 6% reduction in error. Although our absolute boost is smaller, our baseline accuracies are higher, so their finding is similar to the error reduction we found, which corresponds to 6%, 8%, and 7% for *Zooglider*, *ZooScan*, and *UVP5*, respectively (note that the additional error reduction from including our ancillary images is 12%, 8%, and 16%, respectively; Table 2). Yosinski et al. concluded that “initializing with transferred features can improve generalization performance even after substantial fine-tuning on a new task, which could be a generally useful technique for improving deep neural network performance” and we find their maxim extends to multiple rounds of transfer learning with ancillary plankton images.

Our *UVP5* training set is approximately an order of magnitude smaller than either of our other two plankton datasets, and it showed the largest benefit of transfer learning from ImageNet, both in terms of accuracy gain (1 percentage point) and improvement in training time (one third the number of epochs required). The largest accuracy gain from multistage transfer learning also occurred when with the *UVP5* image set (2 percentage points). This is a notable result because newly

developed validation datasets nearly always begin small. The gain to *Zooglider* and *ZooScan* image classifications, while smaller, can still be of appreciable benefit in scientific studies and for specific plankton categories.

All three of our plankton image datasets are relatively unbalanced. If fully balanced, each class would be represented in $\sim 1.5\text{--}3\%$ of the images. Each of our datasets has a single class, detritus, which occurs disproportionately frequently, although much less so that typically observed in field images, where the frequency of occurrence is often over 90%. Plankton datasets could be more balanced by doing one or more of the following: augmenting rarer classes (e.g., Li et al. 2021), subsampling detritus (e.g., Lee et al. 2016), or eliminating or combining rare classes. Johnson and Khoshgoftaar (2019) survey implementations of all three, and find inconclusive results. For our plankton images, we believe augmentation benefits all classes. Subsampling reduces the overall number of training images, running counter to the general finding of more images being better. Rather than combining rarer classes, they often need to be preserved because they can be of high scientific interest. We performed some initial investigation into obtaining a more balanced dataset by decimating the amount of detritus. Based on a small sample size (not shown) this approach seemed to have a detrimental effect on accuracy for the detritus class but did not improve accuracy on rare classes. González et al. (2017) argue that the dataset should reflect the underlying population, with which we agree.

There are three additional practical reasons why we believe that the unbalanced nature of our datasets is appropriate for our use case. First, we are processing entire batches of images from our instruments, so the proportions in our datasets are roughly similar to the proportions at which we will continue to acquire data in the future. Second, balancing the dataset is intended to assist with performance on rare classes, but when we further analyzed the models’ performance by examining confusion matrices, we found that the model still shows some skill on rare classes and transfer learning helps with accuracy on rare classes. Finally, our results show that exposing the network to a larger number of images is beneficial (even if originating from different sources), so it would be inconsistent with that finding for us to discard images.

Recommendations

In summary, our recommendations for training a CNN to classify plankton images begin with assembling as many annotated plankton images as possible, even if images are from seemingly disparate sources. We recommend not expending effort to align the datasets, as simply combining them provided better results. We recommend selecting a network of a size that has achieved reasonably good results on similarly sized datasets in other domains, then finding a pretrained model of that size on ImageNet or a similar source. When performing transfer learning on the combined dataset, allow all layers of the network to be retrained, not merely the fully connected layers. Finally, conduct a brief set of hyperparameter searches during a second round of

transfer learning, again allowing all layers of the network to retrain, during which the combined model is retrained on only the target plankton images. These results should be generalizable to other types of image classification.

References

- Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database, p. 248–255. *In* IEEE Conference on Computer Vision and Pattern Recognition. IEEE. doi:10.1109/cvpr.2009.5206848
- Ellen, J. S., C. A. Graff, and M. D. Ohman. 2019. Improving plankton image classification using context metadata. *Limnol. Oceanogr.: Methods* **17**: 439–461. doi:10.1002/lom3.10324
- Girshick, R., J. Donahue, T. Darrell, and J. Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, p. 580–587. *In* IEEE Conference on Computer Vision and Pattern Recognition. IEEE. doi:10.1109/cvpr.2014.81
- González, P., E. Álvarez, J. Díez, Á. López-Urrutia, and J. J. del Coz. 2017. Validation methods for plankton image classification systems. *Limnol. Oceanogr.: Methods* **15**: 221–237. doi:10.1002/lom3.10151
- Gorsky, G., and others. 2010. Digital zooplankton image analysis using the ZooScan integrated system. *J. Plankton Res.* **32**: 285–303. doi:10.1093/plankt/fbp124
- Guo, C., B. Wei, and K. Yu. 2021. Deep transfer learning for biology cross-domain image classification. *J. Control Sci. Eng.* **2021**: 2518837. doi:10.1155/2021/2518837
- He, K., X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition, p. 770–778. *In* IEEE Conference on Computer Vision and Pattern Recognition. IEEE. doi:10.1109/cvpr.2016.90
- Johnson, J. M., and T. M. Khoshgoftaar. 2019. Survey on deep learning with class imbalance. *J. Big Data* **6**: 27. doi:10.1186/s40537-019-0192-5
- Lee, H., M. Park, and J. Kim. 2016. Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning, p. 3713–3717. *In* 2016 IEEE International Conference on Image Processing (ICIP). IEEE. doi:10.1109/ICIP.2016.7533053
- Li, Y., J. Guo, X. Guo, Z. Hu, and Y. Tian. 2021. Plankton detection with adversarial learning and a densely connected deep learning model for class imbalanced distribution. *J. Mar. Sci. Eng.* **9**: 636. doi:10.3390/jmse9060636
- Lumini, A., L. Nanni, and G. Maguolo. 2023. Deep learning for plankton and coral classification. *Appl. Comput. Inform.* **19**: 265–283. doi:10.1016/j.aci.2019.11.004
- Martinez, M. 2019. Opening Remarks and Awards of the Record-Breaking 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society. Retrieved October 2023 from <https://www.computer.org/publications/tech-news/events/ieee-cvpr-conference-on-computer-vision-and-pattern-recognition-2019-awards-records>.
- Mitra, R., and others. 2019. Automated species-level identification of planktic foraminifera using convolutional neural networks, with comparison to human performance. *Mar. Micropaleontol.* **147**: 16–24. doi:10.1016/j.marmicro.2019.01.005
- Ohman, M. D., R. E. Davis, J. T. Sherman, K. R. Grindley, B. M. Whitmore, C. F. Nickels, and J. S. Ellen. 2019. *Zooglider*: An autonomous vehicle for optical and acoustic sensing of zooplankton. *Limnol. Oceanogr.: Methods* **17**: 69–86. doi:10.1002/lom3.10301
- Orenstein, E. C., and O. Beijbom. 2017. Transfer learning and deep feature extraction for planktonic image data sets, p. 1082–1088. *In* IEEE Winter Conference on Applications of Computer Vision. IEEE. doi:10.1109/WACV.2017.125
- Paszke, A., and others. 2019. PyTorch: An imperative style, high-performance deep learning library, p. 8026–8037. *In* Proceedings of the 33rd International Conference on Neural Information Processing Systems. Association for Computing Machinery.
- Picheral, M., L. Guidi, L. Stemmann, D. M. Karl, G. Iddaoud, and G. Gorsky. 2010. The Underwater Vision Profiler 5: An advanced instrument for high spatial resolution studies of particle size spectra and zooplankton. *Limnol. Oceanogr.: Methods* **8**: 462–473. doi:10.4319/lom.2010.8.462
- Rodrigues, F. C. M., N. S. T. Hirata, A. A. Abello, L. T. De La Cruz, R. M. Lopes, and R. Hirata Jr. 2018. Evaluation of transfer learning scenarios in plankton image classification, p. 359–366. *In* Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2018), v. **5**. SciTePress. doi:10.5220/0006626703590366
- Russakovsky, O., and others. 2015. ImageNet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**: 211–252. doi:10.1007/s11263-015-0816-y
- Sun, C., A. Shrivastava, S. Singh, and A. Gupta. 2017. Revisiting unreasonable effectiveness of data in deep learning era, p. 843–852. *In* Proceedings of the IEEE International Conference on Computer Vision. IEEE. doi:10.1109/ICCV.2017.97
- Thrun, S., and L. Pratt. 1998. Learning to learn. Springer Science & Business Media. doi:10.1007/978-1-4615-5529-2
- Yosinski, J., J. Clune, Y. Bengio, and H. Lipson. 2014. How transferable are features in deep neural networks? p. 3320–3328. *In* Proceedings of the 27th International Conference on Neural Information Processing Systems. Association for Computing Machinery.

Acknowledgments

We thank the Gordon and Betty Moore Foundation for initial support, the U.S. National Science Foundation for support via grants OCE-2243190 and OCE-2224726, and the US Department of Defense for support via a Scholarship for Service Program grant from the SMART SEED Program. Data for analysis and some computation time provided as a contribution from the NSF-supported *California Current Ecosystem* LTER site. Some computation time provided by the DoD High Performance Computing Modernization Program. Annotations provided in part by the Scripps

Ellen and Ohman

Beyond transfer learning

Institution of Oceanography's Pelagic Invertebrate Collection and by
Tristan Biard.

Submitted 16 February 2024

Accepted 14 August 2024

Conflict of Interest

Associate editor: Tammi Richardson

Authors have no conflict of interest.

Supplemental Information

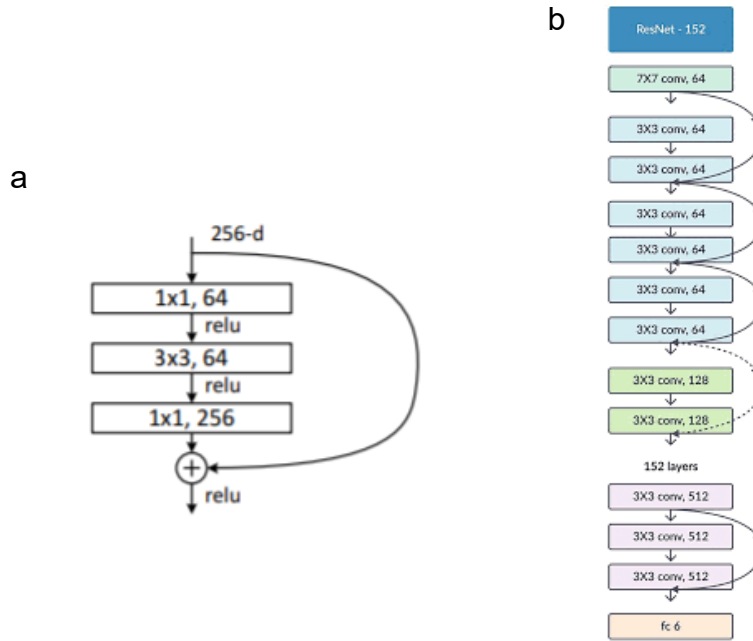


Fig. S1: (A) “Residual ‘Bottleneck’ Building Block,” which facilitates deeper networks while limiting the number of free parameters. (B) An abbreviated view of the whole network (ResNet – 152). ReLU = Rectified Linear Unit; Conv. = Convolution; fc = Fully Connected. See text for explanation. Reproduced from He et al. (2016) (permission being requested).

<u>Aligned Group Name</u>	<u>ZOGLIDER</u>	<u>UVP5</u>	<u>ZOOSCAN</u>
	N=58	N=42	N=30
Acantharia	Acantharia with Large Tests Acantharia Sun-like	Acantharia Acantharia-like	
Appendicularia	Appendicularia <i>Fritillaria</i> without House Appendicularia <i>Fritillaria</i> with House Appendicularia without House Appendicularia with House	Appendicularia Body Appendicularia House	Appendicularia
Artifacts		Artifact Badfocus Artifact Bubble	Badfocus and Artifacts Bubbles
Chaetognatha	Chaetognatha	Chaetognatha	Chaetognatha
Cladocera	Cladocera		Cladocera
Cnidaria+Ctenophora	Ctenophora Ctenophora Velamens Hydromedusae Narcomedusae Hydromedusae Trachymedusae Hydromedusae Trachymedusae Large Siphonophora	<i>Beroe</i> Cnidaria Ctenophora Hydrozoa Siphonophora <i>Solmaris</i>	Cnidaria + Ctenophora

Collodaria	Collodaria	Colonial Collodaria	
Copepoda	Copepoda Others Copepoda <i>Oithona</i>	Copepoda Copepoda-like Eucalanidae	Copepoda Calanoida Copepoda Harpacticoida Copepoda <i>Oithona</i> -like Copepoda Others Copepoda Poecilostomatoida Copepoda Eucalanidae
Crustacea Others	Amphipoda	Crustacea	Crustacea Others Amphipoda
Detritus	Detritus	Detritus-Fiber	Detritus
Diatoms	Diatoms High Concentrations Diatoms without Spines Diatoms with Spines	Diatoms	
Doliolida + Salpida	Doliolida + Salpida	Doliolida Salpida	Doliolida Salpida
Echinodermata	Echinodermata Larvae	Echinodermata	Echinodermata Larvae
Euphausiacea	Euphausiacea Furcilia	Eumalacostraca	Euphausiacea + Decapoda
Multiples		Double Sphere	Multiples
Nauplii + Calyptopes	Nauplii + Calyptopes		Nauplii
Ostracoda	Ostracoda	Ostracoda	Ostracoda

Others

Polychaeta

Polychaeta

Pteropoda

Pteropoda

Pyrosomata

Rhizaria

Phaeodarea Geodesic
Phaeodarea Oblong
Phaeodarea Pin Cushions
Phaeodarea Sphere with Small Spines
Phaeodarea with Branches
Foraminifera

Spheres

Spheres Black
Spheres White

Others

Annelida
Tomopteris

Pyrosomata

Unknown Phaeodaria
Aulacantha
Aulosphaeridae
Castanellidae
Coelodendridae
Foraminifera
Rhizaria-like

Spheres Dark

Others

Polychaeta

Pteropoda + Atlantidae

Pyrosomata

Rhizaria

Groups identified only for a single instrument

UNIQUE to ZOGLIDER

Ceratium
Comets
Dense Background
Dense Fibers
Disks
Edges
Fluffs Black
Overturns
Quasispheres
Spheres Dark Center

UNIQUE to UVP5

Elongated Stick
Feces
Solitary Black

UNIQUE to ZOOSCAN

Bryozoa Larvae
Eggs
Heteropoda

Spindles
Star Spines
Tentacles
Tentacles with White Streaks
Threads
Translucent Clubs
Translucent Spheres
Unknown
V-shaped with Horizontal Line
Walnuts with Noses
Worms Others

34

35 **Table S1.** Listing of individual class names for each of three instruments (*Zooglider*, UVP5, Zooscan) and the corresponding aligned
36 group names used for aligned trials. Aligned trials used 50 total classes (shown in boldface): 23 multi-instrument groups plus 27
37 unique classes identified by a single instrument. Numbers beneath instrument name indicate the number of originally identified class
38 names.