Counting Fish with Temporal Representations of Sonar Video

Kai Van Brunt[1,†] Justin Kay[1] Timm Haucke[1] Pietro Perona[2] Grant Van Horn[3] Sara Beery[1]
[1]MIT [2]Caltech [3]UMass Amherst

Abstract

Accurate estimates of salmon escapement—the number of fish migrating upstream to spawn—are key data for conservation and fishery management. Existing methods for salmon counting using high-resolution imaging sonar hardware are non-invasive and compatible with computer vision processing. Prior work in this area has utilized object detection and tracking based methods for automated salmon counting. However, these techniques remain inaccessible to many sonar deployment sites due to limited compute and connectivity in the field. We propose an alternative lightweight computer vision method for fish counting based on analyzing echograms—temporal representations that compress several hundred frames of imaging sonar video into a single image. We predict upstream and downstream counts within 200-frame time windows directly from echograms using a ResNet-18 model, and propose a set of domain-specific image augmentations and a weakly-supervised training protocol to further improve results. We achieve a count error of 23% on representative data from the Kenai River in Alaska, demonstrating the feasibility of our approach.

1. Introduction

Accurate salmon population monitoring enables data-driven fishery management and conservation. In particular, fishery managers and conservationists are interested in salmon *escapement*: the abundance of migrating salmon returning from the sea that successfully spawn. Several methods exist for monitoring migrating salmon (see Sec. 2). Sonar-based monitoring has recently grown in popularity due to its non-invasive nature and ability to collect data at high temporal resolution under a variety of conditions. However, sonar cameras produce large amounts of data—in some cases over 30GB of data a day [1]—and reviewing this data is time-intensive for technicians, with no existing alternative that generalizes across sites.

Computer vision has the potential to more efficiently and accurately analyze sonar video for escapement monitoring. Prior work has introduced automated approaches based on object detection and multi-object tracking [5]. These approaches achieve counting errors of under 10%; however, they rely upon processing each video frame independently with deep networks (*e.g.* YOLOv5m with 21.2M params [5]), making them currently unsuitable for deployment in locations with limited compute and connectivity.

In this paper, we explore an alternative approach to automated salmon counting in sonar video that harnesses a temporal representation called an *echogram*. Echograms compress a multi-beam sonar video into a 2D image (see Fig. 1). The x-axis of an echogram represents time. At each x-value, a column vector represents a compressed view of an entire frame of video. In this column vector, the pixel intensity at each y-value corresponds to the maximum intensity across all sonar beams at the corresponding range. If fish are present, this will result in a noticeable visual signature. Sonar technicians use these echogram visualizations during data review, both to identify temporal regions of interest and to cross-check challenging counts.

We propose and evaluate the feasibility of a method for analyzing echograms with computer vision to directly predict fish counts, providing a low-compute alternative to object detection and tracking pipelines. Our method takes as input a 200px wide echogram image and predicts the number of fish moving upstream or downstream during the corresponding timeframe, thus requiring only a single forward pass every 200 frames through a lightweight backbone (*e.g.* a ResNet-18 with 11.7M params [3]) to compute counts. We further propose a set of domain-specific image augmentations as well as a weakly-supervised training protocol that incorporates annotations generated ahead of time by an object detector and tracker.

Our initial model achieves counting error rates of 23% on a validation set that is in-distribution with respect to the training set and 30.7% on an out-of-distribution test set, nearly matching initial proofs of concept for much more computationally expense tracking-by-detection approaches [7]. We perform quantitative and qualitative analyses to identify challenges in echogram-based approaches as well as promising areas for future work.

[†]Correspondence to kav@mit.edu

2. Related work

Salmon escapement monitoring. Several methods exist for monitoring salmon escapement, including weirs, counting towers, and various sonar hardware. We include a broader overview of these methodologies in the supplemental material. In this paper we focus on a relatively new generation of sonar hardware known as *imaging sonar* that produce high-resolution videos using multi-beam acoustic hardware. Imaging sonar can be accurately analyzed to count and measure salmon by both human technicians [1] as well as computer vision systems [5].

Computer vision for salmon monitoring. Existing computer vision approaches to salmon counting utilize tracking-by-detection to first perform object detection on individual frames and then link together predicted bounding boxes into trajectories [5, 8]. Once these tracks are determined, different heuristics may be used to determine fish counts. While such approaches produce accurate counts when the training river and testing river are the same, they struggle on out-of-distribution test data sourced from *e.g.* different rivers or different environmental conditions than the training data [4–6]. Another key challenge for real-world deployment of these models is their compute requirements, as fish counting technicians are often stationed in remote locations with only consumer laptops; on such hardware, even efficient object detection and tracking techniques like YOLO [9] and SORT [2] are a severe processing bottleneck. Our method aims to enable more efficient inference by bypassing frame-by-frame video analysis through compressed temporal representations called echograms.

3. Method

3.1. Echogram generation

We begin with sonar video files in the ARIS format [10]. Each file represents 10-20 minutes of continuous sonar footage which may or may not contain fish. At each range (radial distance from the sonar camera) a certain angular span is sampled, outputting a pixel intensity corresponding to the strength of the echo received.

To generate the echogram, we apply successive iterations of background subtraction to each frame of the ARIS file, as in Fig. 1. In each application of background subtraction, after taking the mean frame across the ARIS file, only the pixels of each frame exceeding a threshold value α above the corresponding pixel of the mean frame are kept.

First we apply background subtraction to the raw frame with a low threshold value α_0. Then OpenCV's ConnectedComponentsWithStats function obtains all connected components in the new image which are larger than a size threshold scaled by range. Finally, we apply background subtraction once more with a threshold value α_1 within these components and α_2 outside these components such that $\alpha_0 < \alpha_1 < \alpha_2$. The values of each threshold are tuned by trial and error until a qualitatively acceptable echogram is produced, and we use those same parameters for all data.

This version of the clip, with each frame cleaned of background noise, is used for echogram generation. Each frame of shape (number of samples along range) $\times$ (number of beams) is collapsed into a column of height (number of samples along range), each pixel of the column corresponding to the maximum intensity at that range of the various beams. A second image channel stores the lateral position of that maximum intensity point, normalized between 0 and 1. Concatenating these columns together gives the full 2D echogram, of shape (number of samples along range) $\times$ (number of frames in video).

3.2. Computer vision model

We train a computer vision model in the PyTorch Lightning machine learning framework to predict left and right counts for echogram images. We finetune a ResNet18 model pretrained on ImageNet with a final fully connected layer that contains two outputs corresponding to left and right counts. We use a ReLU activation function after the final layer (since counts must be non-negative) and optimize for mean squared error. We use an input size of 200px by 800px, learning rate of 1e-5 using Adam optimization, batch size of 256, and train for a maximum of 100 epochs on a single NVIDIA A100 GPU with early stopping based on KL-val (see Sec. 4.1) performance.

4. Dataset and metrics

4.1. Data collection and annotation

We generate echograms for the Caltech Fish Counting dataset (CFC) from [5]. We use the default training and validation sets, "KL-train" and "KL-val" from the **l**eft bank of the **K**enai River in Alaska, and we also test on one out-of-distribution test set, "KR" from the Kenai **r**ight bank. In total, this gives us 481 KL-train images, 66 KL-val images, and 406 KR test images. We refer to the ground truth count labels for CFC as **strong labels** in our experiments.

We also generate additional **weak labels** on a set of previously-unlabeled ARIS files collected from the same camera locations as the KL-train and KL-val sets. These weak labels are generated by the public detector and tracker pipeline released with CFC [5]. We label counts in the same way as [5]: a fish whose trajectory start and end are on opposite sides of a vertical line drawn through the center of the frame is counted as either an*left* or *right* traveling fish, based on the relative start and end points of the trajectory. We ensure there is no overlap between the KL validation set and the detector-tracker annotated training or validation set. In total, we generated weak labels using this pipeline for 33,437 images from the KL location.

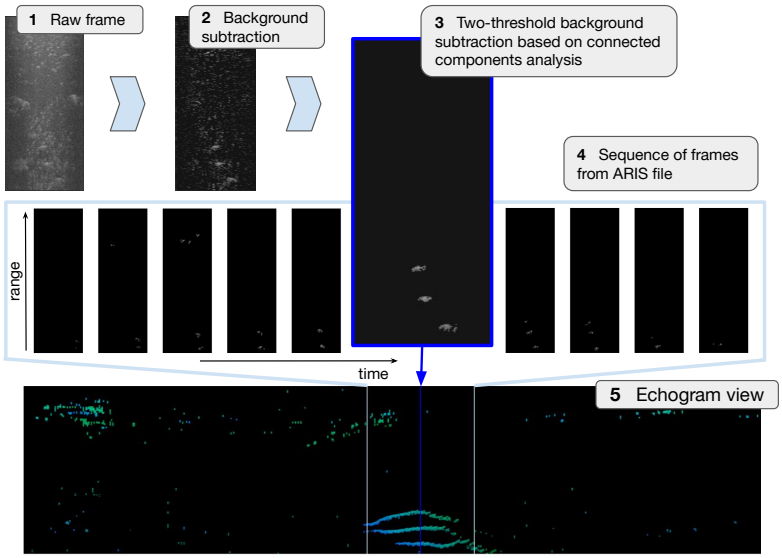

Figure 1. Clockwise: 1) a frame of the raw ARIS file; 2) the same frame after applying background subtraction with a minimum positive threshold on each pixel intensity of $\alpha_0 = 10$ above the mean frame; 3) the same frame after applying connected components analysis, applying background subtraction with a threshold of $\alpha_2 = 127$ outside the largest connected components and a threshold of $\alpha_1 = 35$ inside the largest connected components. 4) Selected frames from a time range in an ARIS file and 5) the same length of time displayed in echogram view, where the color corresponds to the lateral position of the brightest pixel.

There is a large imbalance between leftward and rightward moving fish, since the data is collected to monitor salmon migrating upstream. We orient all clips such that right-moving fish travel upstream and left-moving fish travel downstream, to make the model invariant to the physical upstream direction.

4.2. Metrics

To evaluate model performance we use the normalized Mean Absolute Error (nMAE) as in prior work [5]:

$$\text{nMAE} = \frac{\sum_{i=0}^{N} E_i}{\sum_{i=0}^{N} \hat{z}_i} \quad (1)$$

where N is the number of clips, $\hat{z}_i$ is the target number of counts on the ith clip, and the error E_i is the sum of absolute errors on left and right counts on the ith clip. We also report nMAE for left and right counts separately.

5. Experiments

Our best-performing model achieves an overall nMAE on KL-val of 23% and 30.7% on KR (Tab. 1).The count error on downstream-moving fish is especially high (Tab. 1), due partly to an extreme class imbalance between downstream- and upstream-moving fish in all training sets. In addition, the model systematically predicts lower counts than ground truth in clips with large numbers of fish (Fig. 2), where separate tracks on an echogram may overlap and become difficult to distinguish.

These error rates are higher than state-of-the-art detector-tracker pipelines for salmon counting: Kay et al. [5] achieved 4.9% error on KL-val and 11.8% on KR using a YOLOv5m detector, and reduced these errors to 3.3% error on KL-val and 3.7% error on KR using a more complex input representation. However, our results are comparable to initial results published by the same team [7] that reported counting error rates of 19.3%, indicating the potential to improve our results in future work.

Our experiments in Tab. 1, Tab. 2, and Tab. 3 demonstrate that our model's performance is improved by incorporating both weak and strong labels during training, tuning the echogram generation parameters, and by applying domain-specific data augmentations during training. We ablate these contributions next.

5.1. Training data

	KL-val nMAE (%) ↓			KR nMAE (%) ↓		
Training set	Total	Down	Up	Total	Down	Up
KL-train ([5], 481 imgs)	42.1	100.0	39.4	61.0	102.7	57.9
KL-weak (Ours, 33k imgs)	44.3	112.5	41.1	34.8	96	30.3
KL-train + KL-weak	**23.0**	**37.5**	**22.3**	**30.7**	**96.0**	**25.8**

Table 1. Dataset choice vs performance on KL-val and KR, split by downstream ("down") and upstream ("up") moving fish. Training with strong and weak labels improves over both a small dataset of strong labels only, and a large, diverse dataset of weak labels only.

We train the model on three different datasets: one composed of weak labels only; one composed of strong labels only; and one composed of a mixture of all weak labels and strong labels. In Tab. 1, the drastic difference between nMAE for the model trained only on strong labels vs the mixture of weak and strong labels (about a 20% improve-

ment for KL-val and 30% improvement for KR) indicates that training on a large, diverse dataset improves the model despite the potential inaccuracies present in the weak labels. The inclusion of strong labels, which make up less than 2% of the total dataset size, also significantly improves model performance compared to the model trained on weak labels only, especially on the in-distribution test set (KL-val).

5.2. Echogram generation parameters

Echogram params				nMAE (%) ↓
α_0	α_1	α_2	*size_thresh*	KL-val
0	0	0	0	84.7
20	0	0	0	36.6
20	**40**	**60**	**100**	**23.0**
20	40	100	120	37.2

Table 2. Echogram generation parameters vs performance on KL-val and KR for models trained and validated on a mixture of weak and strong labels. The model performs best at some intermediate setting where a balance is achieved between filtering out background noise and preserving information about fish tracks.

When generating the echogram slices used as input and test data for the model, thresholds for initial background subtraction, secondary background subtraction, and filtering based on size can be tuned. Higher thresholds lead to information loss but also produce a cleaner, less noisy signal for the model. Testing different sets of thresholds as in Tab. 2 shows that an intermediate setting is ideal: the model benefits from some filtering of noise but is negatively affected by cutting background signal too aggressively.

5.3. Data augmentations

We explore various data augmentation strategies, informed by the specifics of the echogram image domain, in Tab. 3.

Vertical flip. Flipping the entire image across the horizontal axis improves nMAE for all model setups.

Naive horizontal flip. Regardless of the set of labels the model is trained on, nMAE worsens when a horizontal flip augmentation is applied, which flips the entire image. In all training and validation sets, a class imbalance between upstream- and downstream-traveling fish exists: during spawn season, many more fish are swimming upstream than downstream. In KL-val, 175 fish are swimming upstream while only 8 are swimming downstream. In addition, the upstream and downstream motion patterns of fish are different due to the direction of the river current. This naive horizontal flip augmentation thus both obscures the true distribution of upstream vs. downstream counts and does not accurately capture the motion of the fish in the opposite direction, suggesting that a different method is needed to robustly classify downstream-swimming fish.

Realistic horizontal flip. This transformation reflects the image across the horizontal axis and then inverts the lateral position channel to match the original, pre-reflection direction of fish motion. This improves nMAE across all model setups but does not improve the left-right class imbalance.

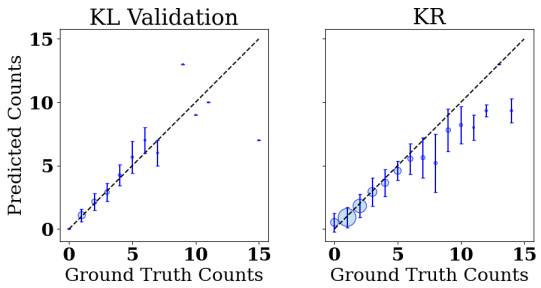

Figure 2. Mean and standard deviation of total predicted counts vs total ground truth counts per clip on the KL-val and KR test sets. Size of the dot corresponds to the number of images with the associated ground truth count. The model systematically predicts lower counts than ground truth for KR clips with large numbers of fish, where tracks of distinct fish may overlap and become difficult to distinguish on the echogram.

Superposition. Two echograms are superposed, displaying at each point the intensity and color of the brightest pixel; the target counts are added together. This augmentation has mixed effects on performance, modestly improving training on weak labels while worsening training on strong labels.

Data augmentations				Train set	
V. flip	H. flip	Superpos.	Realistic h. flip	KL-train	KL-weak
				48.1	64.5
•				43.2	49.2
•	•			111.5	53.6
•	•	•		129	49.7
•		•	•	74.3	**37.2**
•			•	**39.3**	44.3

Table 3. KL-val nMAE using either KL-train or KL-weak and ablating data augmentations, using cleanest echogram generation settings. A naive horizontal flip augmentation worsens the performance of the model trained on either dataset, while vertical flip and our domain-specific realistic horizontal flip improve results.

6. Conclusions

We introduce a new method for salmon population monitoring based on an *echogram*, a 2D representation of an entire sonar video clip, that is more computationally efficient than existing methods for determining fish counts which are applied to individual frames of a sonar video. Our initial results are promising: a lightweight ResNet-18 model achieves significant reductions in nMAE which bring us to count errors comparable to proofs of concept of more expensive models, through appropriate dataset selection, echogram generation, and data augmentation.

Future evaluations and iterations on this model should address the class imbalance between upstream- and downstream-moving fish, develop a larger and more diverse validation set, and fine-tune the echogram generation and data augmentation procedures.

Acknowledgements

This material is based upon work supported by: the MIT Climate and Sustainability Consortium Scholars Program, MIT J-WAFS seed grant #2040131, National Science Foundation award #2330423, and Caltech Resnick Sustainability Institute Impact Grant "Continuous, accurate and cost-effective counting of migrating salmon for conservation and fishery management in the Pacific Northwest." Thanks to Erik Young and Suzanne Stathatos for input and discussions, and Bill Hanot for initial conversations on echogram generation.

References

[1] Alaska Department of Fish and Game. Sonar tools: Imaging sonar. https://www.adfg.alaska.gov/index.cfm?adfg=sonar.didson, n.d. Alaska Fisheries Sonar. 1, 2

[2] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 2

[3] 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. 1

[4] Justin Kay, Suzanne Stathatos, Siqi Deng, Erik Young, Pietro Perona, Sara Beery, and Grant Van Horn. Unsupervised domain adaptation in the real world: A case study in sonar video. In *NeurIPS 2023 Computational Sustainability: Promises and Pitfalls from Theory to Deployment*. 2

[5] Justin Kay, Peter Kulits, Suzanne Stathatos, Siqi Deng, Erik Young, Sara Beery, Grant Van Horn, and Pietro Perona. The caltech fish counting dataset: A benchmark for multiple-object tracking and counting, 2022. 1, 2, 3

[6] Justin Kay, Timm Haucke, Suzanne Stathatos, Siqi Deng, Erik Young, Pietro Perona, Sara Beery, and Grant Van Horn. Align and distill: Unifying and improving domain adaptive object detection. *arXiv preprint arXiv:2403.12029*, 2024. 2

[7] Peter Kulits, Angelina Pan, SM Beery, Erik Young, Pietro Perona, Grant Van Horn, and Trout Unlimited Caltech. Automated salmonid counting in sonar data. In *NeurIPS 2020 Workshop on tackling climate change with machine learning*, 2020. 1, 3

[8] Mohammed Yasser Ouis and Moulay Akhloufi. Yolo-based fish detection in underwater environments. *Environmental Sciences Proceedings*, 29(1):44, 2023. 2

[9] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[10] Sound Metrics. Aris sonars. http://www.soundmetrics.com/Products/ARIS-Sonars/, 2024. Accessed: 2024-08-14. 2