LIMNOLOGY
and
OCEANOGRAPHY: METHODS

ASLO

Limnol. Oceanogr.: Methods 2026, e70024

doi: 10.1002/lom3.70024

NEW METHODS

PNNL-TUNAMELT: Toward automating the detection of interactions with marine energy devices using acoustic camera sensors

Theodore Nowak [1]*, **Garrett Staines** [1], **Blerim Abdullai** [2]

[1]Pacific Northwest National Laboratory, Coastal Sciences Division, Sequim, Washington, USA; [2]Department of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, Urbana, Illinois, USA

Abstract

Acoustic cameras, or imaging sonars, are often used to monitor marine energy sites in regions where the water is too dark or turbid for optical sensing. To do so more effectively, scientists are investigating automated detection methodologies to use on these data. However, prior work has found that existing automated detection approaches struggle with the dynamic image background around marine energy devices—such as moving turbine blades. While open-access datasets, methods, and standard evaluation metrics are needed to quickly develop and compare novel automated detection methods, none yet exist for this domain. Using previously collected data, in this work we created a labeled dataset of possible marine life interactions in acoustic camera video around an operating tidal turbine. We call this dataset the Pacific Northwest National Laboratory dataset for Tracking Underwater Nautical Activity around Marine Energy LocaTions or PNNL TUNAMELT dataset. In addition to this dataset, we developed an automated detection pipeline which filters noise from the acoustic camera imagery and then performs object detection to identify possible targets. To analyze our automated detection pipeline, we used a series of common detection and classification metrics. In doing so, we found that our pipeline detected 98% of targets and removed 70% of target-less frames in our dataset. These results illustrate our method's potential utility as an aid to a human analyst tasked with extracting targets of interest from the dataset. Finally, we openly release our labeled dataset and all associated code to support and encourage future work in this domain.

Marine energy devices, such as underwater turbines, convert tidal (Staines et al. 2019) or river currents (Bevelhimer et al. 2017) into electricity, and are currently in the research, development, and testing phases in the United States (Kilcher et al. 2021). Environmental concerns around marine energy technology include the risk of collision to fish and other marine life (Copping and Hemery 2020). At these deployments, regulators can require monitoring efforts to inform assessments of collision risk concerns, and when doing so, imaging sonars (Cotter and Staines 2023) are a sensor often used to observe interactions around turbines (Bevelhimer et al. 2017; Staines et al. 2019; Viehman and Zydlewski 2015) and turbine infrastructure (Williamson et al. 2021), especially when optical cameras are rendered ineffective due to low light or water turbidity.

Monitoring campaigns around marine energy devices with imaging sonars can be weeks or months long during which terabytes of data may be accrued. Presently, the processing of these data to identify rare interaction events requires human review and is laborious (Baumgartner et al. 2006; Eggleston et al. 2020). During this review, capturing every possible event is critical as a record of all interactions is needed to inform regulatory decision-making at the current stage of marine energy device testing. To ensure all possible interactions are captured, these large datasets cannot be subsampled for risk of missing an event of interest (Matzner et al. 2017). For automated

*Correspondence: theodore.nowak@pnnl.gov

Associate editor: Craig Lee

detection methods to be successful in this domain, they need to be capable of processing long-duration datasets with high true positive (TP) rates to robustly flag these events of interest.

The processing of data acquired from acoustic cameras has mostly been applied for fishery resource management decision-making and primarily performed manually by human reviewers (Grote et al. 2014; Holmes et al. 2006). During this process, targets are detected and identified (Egg et al. 2018; Gurney et al. 2014; Key et al. 2016; Melvin and Cochrane 2015; Smit et al. 2016; Tušer et al. 2014), the abundance of fish communities or other biomass metrics is documented (Braga et al. 2022; Burwen et al. 2010; Hightower et al. 2013; Key et al. 2016; Xie and Martens 2014), and fish behavior is described (Rakowitz et al. 2012; van Keeken et al. 2020; Viehman and Zydlewski 2015). Data cleaning, dataset curation, and automated detection methods have been primarily explored for these fishery applications (Fernandez Garcia et al. 2023; Helminen and Linnansaari 2021; Kay et al. 2022; McCann et al. 2018). While most of these works did not evaluate data that contained a moving background, those that did have found that current approaches fail in such circumstances (Capoccioni et al. 2019; Han et al. 2009).

To automatically detect targets in imagery, object detectors are used. Filters are also often applied to simplify the detection problem when consistent and characterizable noise is present in the imagery. Most object detection approaches have been designed and studied on optical camera imagery or video due to the prevalence of these data. While imagery generated by imaging sonars is video-like and can be ingested by image-based object detectors, the patterns of data described in their images are different. The pixel values in acoustic camera imagery represent intensities of acoustic reflection, and the resolution of their video is lower and varies with both range and beamwidth. Additionally, there is increased and more complex noise in acoustic camera imagery due to multiple beam transmissions and time-resolved signals that can overlap, creating noise artifacts. Finally, the viewing perspective of imaging sonars is potentially non-intuitive, where imagery rendered is often orthogonal to the range and beamwidth of the unit's beam array. As such, unique image processing pipelines must be developed, and existing, learned, image-based object detectors must be retrained or fine-tuned on datasets of labeled acoustic camera imagery in order to be performant on it.

In the marine energy domain, existing approaches that perform object detection fail to accurately detect and track marine life around moving tidal turbines (Hasselman et al. 2020). When monitoring fish interactions around tidal turbines, the sonar beam array ensonifies the turbine and its blades in order to capture animal interactions (Cotter and Staines 2023). This introduces a moving, acoustic reflector into the field of view, including some but not all beams in the array. Approaches reliant on subtracting static background from target motion to perform object detection and recognition necessarily fail (Gillespie et al. 2023).

Efforts to produce performant object detectors suitable for this domain are hindered by a paucity of labeled data and common evaluation metrics with which to develop and compare approaches. These labeled datasets are especially critical to the development and evaluation of performant machine learning approaches on these data. However, no labeled, open-access datasets yet exist for target detection and tracking in acoustic camera video around marine energy devices.

In this paper, we address this need by labeling and openly releasing a primary dataset of Sound Metrics Corporation (SMC), Dual-Frequency Identification Sonar (DIDSON) data originally collected by Viehman and Zydlewski (2015) with bounding box labels around targets in the acoustic camera video. In addition to this dataset, we propose and evaluate a baseline automated detection pipeline that was optimized on the training split of this dataset and evaluate our approach using a series of metrics on the test split, which we propose be used by future works in this space. Rather than solely focusing on a detection approach like some prior work (Kay et al. 2022), we instead split the pipeline into filtering and detection stages. In the filtering stage, we seek to reduce the noise induced by the turbine motion or other conflating effects. In the detection stage, we seek to detect targets in the filtered video output by the filtering stage. In this way and by seeking to never filter out targets of interest, our method accommodates and enables future improvements to detector performance as well as generalization to other application domains.

The goal of our automated pipeline was to reduce the burden on a human analyst tasked with monitoring a marine energy site. As such, we sought to *classify* frames of data as either containing a target or not containing a target (rather than performing *object detection* and accurately estimating target locations in the scene). Therefore, while object detection bounding boxes are given as labels in our dataset, we evaluate our approach as a per-frame classification task. At the outset of this work, we defined a desired performance for our method: greater than 90% target detection (correctly detecting each unique target) and greater than 20% frame removal (identifying frames without targets). Our approach exceeded this goal. When evaluating our method on the unseen test data of our dataset, we were able to achieve 98% target detection and 70% frame removal with our most-performant model.

Contributions

- A labeled, open-access acoustic camera video dataset of targets near an active tidal turbine designed for automated detection development and comparison.[1]
- A novel, open-source automated detection pipeline designed to filter out turbine motion, which achieves 98% target detection and 70% non-target frame removal on the test data.[1]
- An evaluation of our dataset and approach.

[1]Link to the PNNL TUNAMELT dataset and detection code: https://github.com/tsnowak/pnnl-tunamelt.

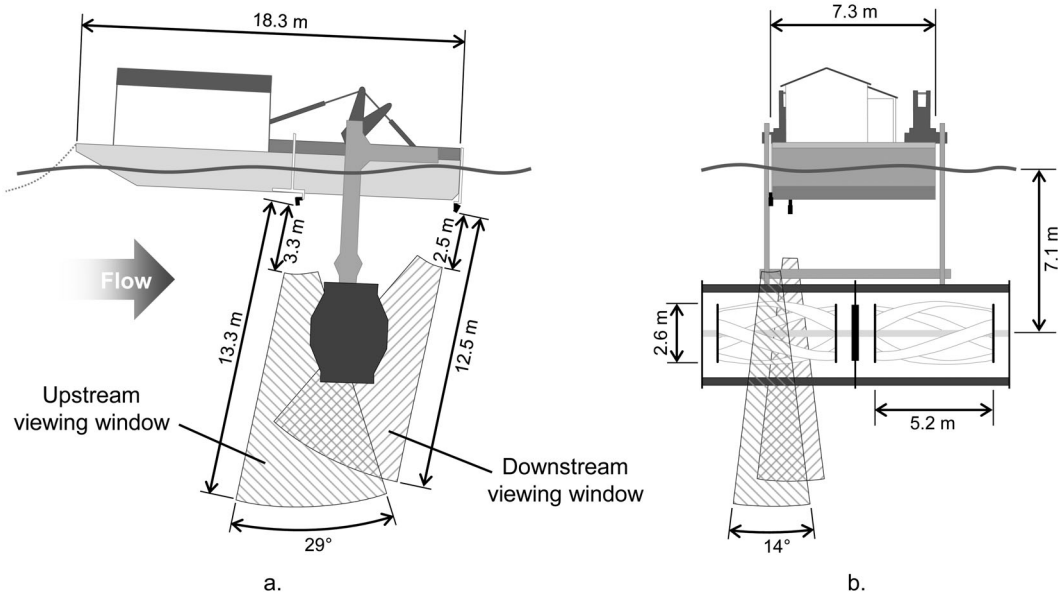

Fig. 1. Illustrations from Viehman and Zydlewski (2015) showing the DIDSON beam array fields-of-view with respect to the turbine.

- A discussion of future improvements to data collection and analysis around tidal and riverine turbines.

Materials and procedure

Dataset acquisition

This study uses samples from a 23-h dataset that was collected using two down-looking SMC DIDSON acoustic cameras attached to a research barge that operated an experimental tidal turbine (Fig. 1) in Cobscook Bay, Maine, USA (Viehman and Zydlewski 2015). The full dataset consists of 15-min data files in SMC DIDSON .ddf format with an approximate frame rate of seven FPS. For specifics of the site, research barge, turbine, and data collection procedures, we refer the reader to Viehman and Zydlewski (2015).

Dataset

The original 23-h dataset was reduced and split into two subsets of data. A training subset or split (training data) consisting of 21 video snippets varying in length from 3 to 67 s, and a testing subset or split (testing data) consisting of seventeen 15-min video files (Table 1).[2] The training data were selected for known target presence as determined by manual inspection, while the larger testing data were chosen by manually inspecting each file for turbine operation, that is, tidal currents high enough to engage blade rotation. Both subsets sought to capture site variability that included fore and aft fields-of-view (FOVs) of the turbine, current speed, tidal stage, and surface-oriented entrained air. All training and testing data were converted to .mp4 video files in cartesian space using the movie export function in the SMC ARISFish software (ARISfish 2020). Each frame of the exported videos contains the acoustic camera image (in cartesian coordinates) with a black background and an overlay of the range from the acoustic camera, in meters. The following settings were applied to the data in ARISFish before exporting: the signal intensity histogram values were set to 0 and 35.1 dB, the Palette color was set to "Deep Blue," effects were set to "None," measure was set to "Geometry," Frame Rate was set to 10 frames per second, and no filters were used (ARISFish manual sec 6.4 and 6.5). Testing data videos were split into video snippets no more than 30 s in length before processing. An example frame from the dataset is in Fig. 2. Statistics about the dataset and splits are in Table 1 and Fig. 3.

Data annotation

All targets identified by a human reviewer in the training and testing data were manually annotated per frame with the closest-fitting bounding boxes. This was done by two annotators—one subject matter expert and an assistant—using the open-source Computer Vision Annotation Toolkit (CVAT 2022). For each target, the annotator placed a bounding box around a target in the first frame in which it appeared and then placed an instance of the same bounding box around the last frame in which the target remained in the FOV. The linear interpolation function built into the video bounding box annotation tool in CVAT generated the bounding boxes for frames between the start and end box. The annotator manually reviewed each frame containing the target to ensure that the bounding box was well-fitting until the target was out of the FOV. Bounding box annotations were exported from CVAT as .xml files in the CVAT-Video version 1.1

[2]Note that no validation split was created for this dataset. In this work, hyperparameters were optimized on the training data. We assume that future works perform *k*-fold cross-validation using the training data or create their own validation set from the training data to evaluate performance during training and parameter optimization.

Table 1. Comparative statistics of the training and testing data.

Subset	# .mp4 video files	Average # of frames	Total # of targets	Percent frames with targets (%)	Median target size (pixels)
Training	21	192	145	17	221
Testing	17	6084	121	1	320

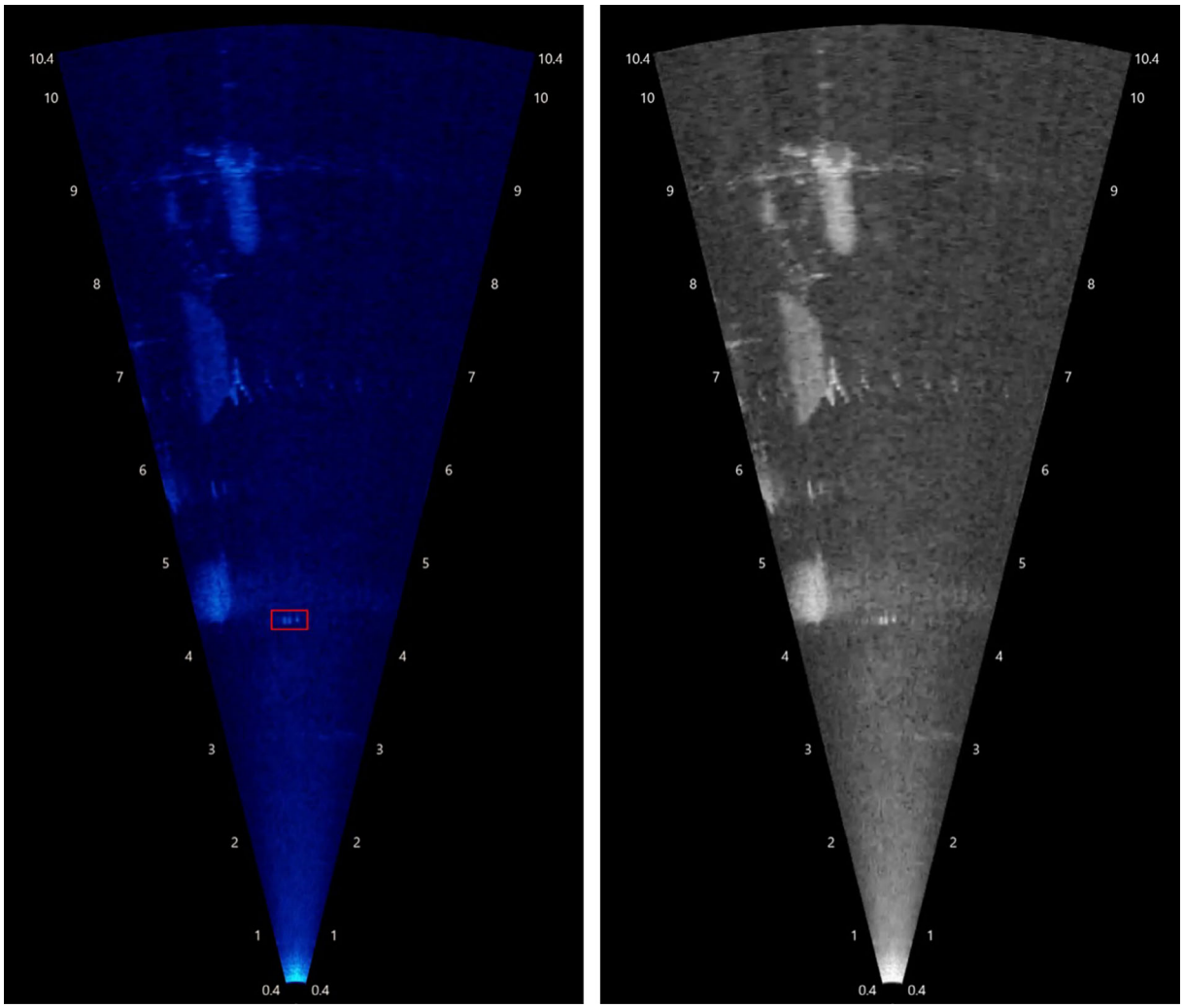

Fig. 2. Left—Example frame of .mp4 video converted from the original SMC DIDSON .ddf with the target labeled (bounding box) in red. Right—the value channel of this frame after it was converted to hue, saturation, and value (HSV) space. Note these frames are the same as those displayed in Fig. 4. Numbers on the sides of the field of view are range values in meters from the SMC DIDSON.

format. These label files, in addition to the .mp4 video files, were used for all pipeline development, refinement, and testing, and are available for public download and use.[1]

Notation

We first define our data and filters in the abstract and describe the notation used in the equations in subsequent sections which comprise our automated detection pipeline.

Consider the set $P = \{x \in W | x \leq 255\}$ where P is the set of possible whole number, pixel values between 0 and 255. All our videos, their frames, as well as the outputs of our filter functions reside in this greyscale image space. We define the videos in our dataset as $X \subset P^{N \times W \times H \times C}$ and the set of frames within said videos X as $X_n \subset \mathrm{P}^{W \times H \times C}$. Here, N is the number of frames in the video, W and H are the width and height of a video frame respectively, and C is the number of channels in

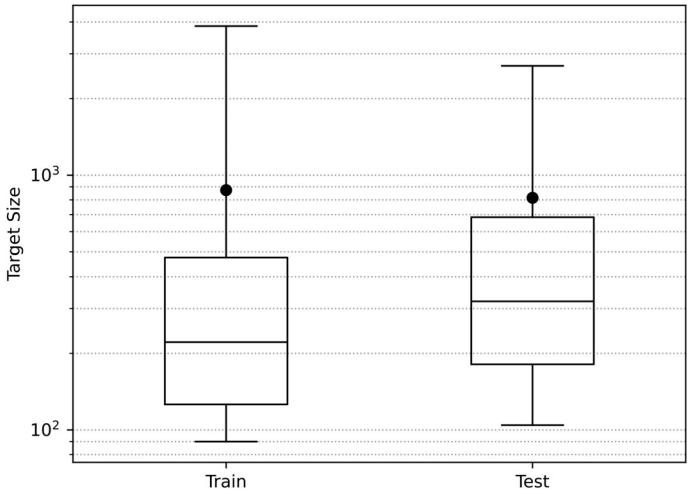

Fig. 3. Box and whisker plots showing the bounding box size distribution of targets in both the train and test set. The *y*-axis is log scaled. The mean is indicated by a dot while the median is indicated by the middle bar of the box. The upper and lower bars of the box represent the third and first quartiles, respectively, while the upper and lower extrema bars (whiskers) are drawn at the 95 and 5 percentile values.

the frame.[3] Filtering methods are applied either to an entire video, F: $X \rightarrow Y$, or per frame, F_N: $X_n \rightarrow Y_n$, where video or per-frame filter outputs are defined as $Y \subset P^{N \times W \times H \times C}$ and $Y_n \subset P^{W \times H \times C}$, respectively. Methods that are unique to each frame are written as F_n: $X_n \rightarrow Y_n$. Individual pixels in a frame of the video are referenced with lowercase indices corresponding to their parent dimension: x_{nwhc}. Many of our methods generate binary masks that are used to zero pixels from frames in the video. These can be thought of as applying the indicator function $1_{F_{wh}}$ at each pixel wh using a given Boolean function F_{wh}, which then returns 1 when F is true and 0 otherwise. We denote the logical negation of this function as $\overline{1}_F$, which returns 0 when F is satisfied and 1 otherwise. The binary masks we generate are either unique to each frame, or consistent across all frames in the video. These masks are written $\Theta \subset Z_2^{N \times W \times H}$ for video-wide masks, or $\Theta_n \subset Z_2^{W \times H}$ for per-frame masks. Some methods generate per-pixel vectors of values $Y_{nwh} \subset \mathrm{P}^K$. We index these vectors with superscripts $y^k_{nwh}, \forall k \in K$. Finally, targets of interest exist in these videos, and we refer to an individual target in video m as $t \in T_m$.

Methodology overview

This work takes the approach of first filtering the video input to remove non-target background objects and noise, then subsequently performing detection and tracking. During filtering, a series of steps—mostly composed of binary mask filters—are calculated to remove features in the scene that confound target detection. These can also be thought of as components of noise that obfuscate the signal of targets in the video. We sought to remove all sources of noise during filtering, but certain filter stages were designed to target specific noise sources or effects. These noise sources were static background, turbine motion, and random speckle. While not explicitly written in each step below, the generated binary mask is multiplied by the output of the previous filtering step to yield the input used in the next step of the pipeline. This filtering simplifies the detection task by zeroing non-target pixels in the video. After such filtering, we applied a relatively simple detection regime to detect and bound targets. OpenCV's implementation of Suzuki (1985) was used to detect clusters of high-intensity pixels and generate per-target bounding box predictions. Cross-frame detection association was then performed to pair similar detections across frames into tracklets (frame-to-frame bounding box pairings that comprise object tracks) to filter out detections that were inconsistent across frames. The sequence of filters, detectors, and intermediate inputs and outputs is visualized via the pipeline in Fig. 4. In the following section, each algorithm that together constitutes our filtering and detection pipeline is described in the order that it is applied in our proposed model.

Filtering methods

Background-removal filter

We first applied a windowed-average, background-removal filter to remove static objects from the original video. As shown in Eq. (1), this approach uses the per-pixel mean μ^K_{wh} and standard deviation σ^K_{wh} to remove pixels per frame that lie outside s standard deviations of their mean as calculated across a window of length K. Intuitively, this only preserves pixels in the current frame that are of significantly larger intensity than their average value in the window. The results of this filter applied to raw video can be seen in the *Background-Removal* stage of Fig. 4.

$$\Theta_{nwh} = x_{nwh} > \mu^K_{wh} + s\sigma^K_{wh} \quad \forall n \in N, w \in W, h \in H \tag{1}$$

Discrete Fourier Transform–based turbine filter

Subsequently, we sought to remove pixels containing the motion of the turbine. To do so—and noting the unique, periodic nature of the turbine—we designed a per-pixel, Discrete Fourier Transform (DFT)–based filter that removes pixels which exhibit significant periodicity within a certain frequency range proportional to the rate of the rotation of the turbine. We define this frequency range proportional to rotation rate as $U \propto R$.

The DFT decomposes a time-varying signal into B frequency bins (π^B) of varying magnitudes ($||\pi^B||$) that describe the original time-series signal in terms of frequency.

[3]Typically, $C = 1$ in this work since we convert our image to *HSV* space and only operate on the *Value* channel.

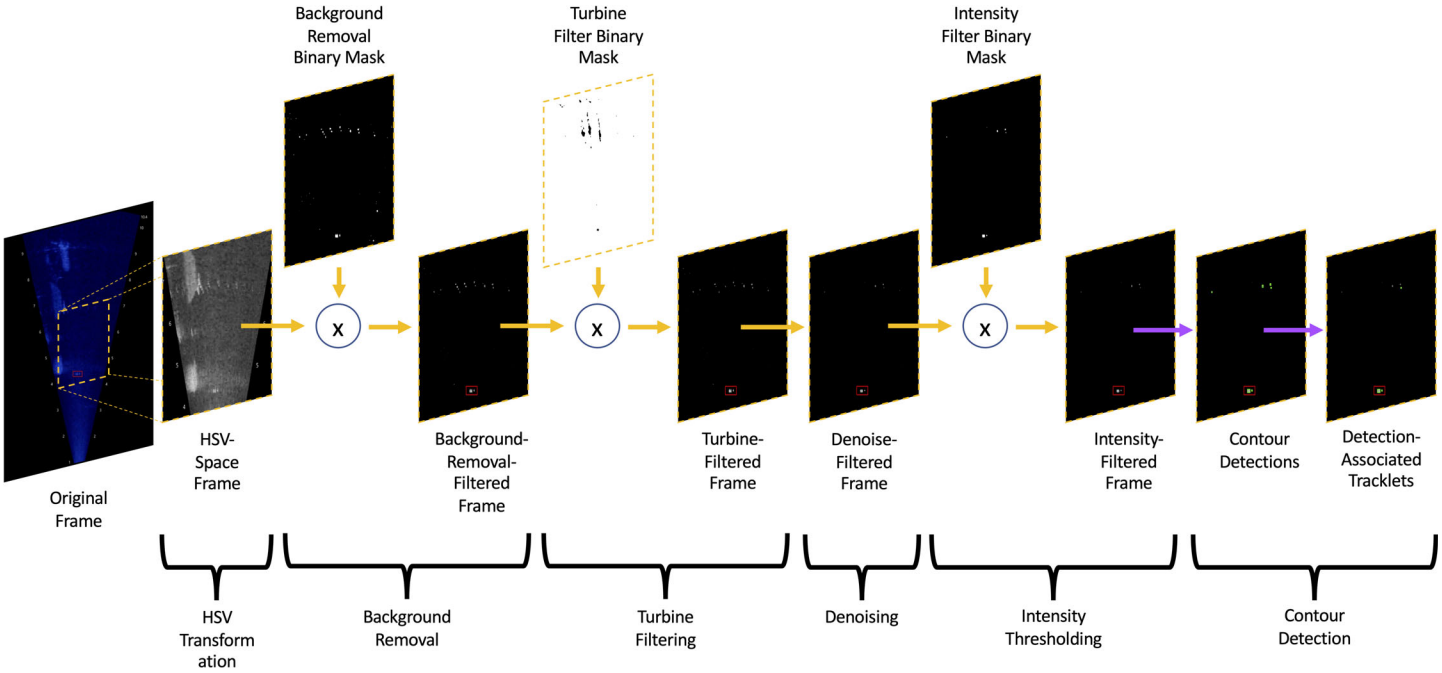

Fig. 4. Illustration of our filtering and detection pipeline. After the original frame, an exploded crop is shown of proceeding filtering steps. Ground-truth bounding boxes are red, while predicted bounding boxes are green. Yellow arrows are used for filtering steps, while purple arrows are used for detection steps. Note that this figure is best viewed in color and at high resolution.

After applying the DFT, we can more easily distinguish signals that exhibit certain frequency patterns. In addition, we can also readily quantify the contribution of each of these frequencies to the overall signal. As such and concretely, we calculated the DFT of each pixel over all time in the video snippet. Having pixels' frequency components and magnitudes, we then removed the pixels whose highest magnitude frequency ($\pi^{\text{argmax}||\pi^B||}$) component contains the rotation rate (U) of the turbine: those that exhibit the periodicity of a rotating turbine more than any other frequency. The results of our turbine filter applied to a background-subtracted video frame can be seen in *Turbine Filtering* stage of Fig. 4.

$$\forall \; w \in W, h \in H$$
$$\Theta_{wh} = \overline{1}_{F_{w,h}}, \quad F(w,h) := U \in \pi_{wh}^{\text{argmax}||\pi^B||} \tag{2}$$

Gaussian Blur image denoising

The resulting video after applying the turbine filter is mostly free of pixels containing background and turbine information. However, given the low resolution and high noise of acoustic cameras, in addition to some errant acoustic reflections from the turbine, there is still information contained in the video other than our targets of interest. Therefore, we applied a 2D Gaussian Blur across each frame to remove spurious, noise-induced signals—the results of which can be seen in the *Denoising* stage of Fig. 4.

Pixel-intensity thresholding

Given that targets are acoustically reflective, we sought to zero any remaining low-intensity background pixels such that only clusters of high-intensity plausible targets and zeros remained. We thus zeroed remaining pixels whose intensities were below a threshold value ζ: $\Theta = X > \zeta$.

Detection methods

Contour detection

After the previous filtering steps, clusters of remaining high-intensity pixels were automatically bounded with boxes to indicate plausible target detections. To do so, we used the contour-based detection method proposed by Suzuki (1985) and now built into OpenCV's computer vision library (Bradski 2000). To refine these detections, we filtered them by size to include only those that are greater than a minimum bounding box area $a_{\min}$, and less than a maximum box area $a_{\max}$. Such bounding boxes can be seen in green in the first segment of the *Contour Detection* stage of Fig. 4.

Detection association

After detecting high-intensity contours in each frame of the video, we sought to refine our detections by filtering out those lacking spatiotemporal consistency. We noted that noise, unlike targets, appears spuriously in a single frame and without consistency in location or size. To filter out this noise, we defined a detection association algorithm to pair detections across frames and identify sequences of detections (tracklets)

to keep only those that are plausible. We calculated this association plausibility score using a set of cost functions that score each bounding box association according to the distance between pairs of box centroids C, and the difference in their shapes S. We performed this association calculation between every box in a window of ω frames around the box's frame to ensure that we associated detections of targets that were detected inconsistently. We did so using a greedy approach—associating the minimum-valued subsequent box with replacement—unlike the typical optimal bipartite graph association as calculated via Kuhn–Munkres to reduce complexity (Kuhn 1955).

Given detection d_j in a given video frame i, let there exist a detection window W of proceeding frames in which we may pair detections to form a tracklet. If detection d_k exists in frames $i+\omega \ \forall\omega \in \{1,\ldots,W\}$ that satisfies $\text{Threshold} \geq S(d_j,d_k)+C(d_j,d_k)$, then let the detection with minimum cost form the detection-associated tracklet pair $(d_j, d_{k_{\min}})$. If no such tracklet exists, then the detection d_j is considered invalid and ignored. The effects of this approach can be seen in the *Detection Association* stage of Fig. 4.

Evaluation metrics

We assessed our method as a binary classifier per frame and evaluated positive or negative predictions based on the detection of at least one target in the frame. Because we consider this a binary-classification problem, we calculated the Average Recall (AR), Average Precision (AP), $F1$-score ($F1$), Percentage of Frames Removed (FR), Target Detection Rate (TDR), and mean of the FR and TDR (MFRTDR) when evaluating the efficacy of our approach. These metrics are described in detail below.

Average recall

Recall (R) is the ratio of correct detections—TPs—to total targets in the video—TPs plus false negatives (FNs). Recall thus captures the percentage of targets that were detected in each video. Average Recall captures the average of this percentage over all M videos in the dataset, and is calculated as follows:

$$\text{AR}_m = \frac{\text{TP}_m}{\text{TP}_m + \text{FN}_m}, \ \text{AR} = \frac{1}{M}\sum_m^M \text{AR}_m \tag{3}$$

Average precision

Alternatively, AP is calculated as the TPs over the TPs plus false positives (FPs) averaged over all videos, as shown in Eq. 4. It therefore captures the precision of the model's guesses when detecting targets.

$$\text{AP}_m = \frac{\text{TP}_m}{\text{TP}_m + \text{FP}_m}, \ \text{AP} = \frac{1}{M}\sum_m^M \text{AP}_m \tag{4}$$

F*1-score*

The $F1$-score ($F1$) is the harmonic mean of recall and precision and thus captures the best mixture of the two. We calculate $F1$ per video and average over all, as shown in Eq. 5.

$$F1 = \frac{1}{M}\sum_m^M \left(2 \times \frac{\text{AP}_m \times \text{AR}_m}{\text{AP}_m + \text{AR}_m}\right) \tag{5}$$

Percentage of frames removed

In addition to the standard metrics above, we consider the percentage of frames removed from a video in accordance with our previously stated goal of aiding human reviewers. This is calculated as the number of frames predicted to be without targets over the total number of frames, averaged over all videos. Here, TN_m is the true negative rate in video m.

$$\text{FR} = \frac{1}{M}\sum_m^M \frac{(\text{TN}_m + \text{FN}_m)}{N_m} \tag{6}$$

Target detection rate

To capture our method's ability to notify an analyst of every target in the video—rather than whether we detect a target in every frame in which it exists (as recall describes)—we calculate the TDR. To do so, we measure the percentage of targets where we have at least one detection in at least one of its frames. We then average this value across all videos to determine the TDR.

$$F := \exists \text{TP}_t \text{ for } t \in T_m$$

$$\text{TDR} = \frac{1}{M}\sum_m^M \frac{1_F}{|T_m|} \tag{7}$$

Mean of the frame removal and target detection rates

Finally, and to consider both FR and TDR, we calculate the mean of these two metrics. We refer to this as MFRTDR:

$$\text{MFRTDR} = \frac{(\text{FR} + \text{TDR})}{2} \tag{8}$$

Hyperparameter search

Each of the aforementioned filters contains free parameters whose values must be selected. To do so, we performed a search over a subspace of plausible values which we identified qualitatively. Following the approach and terminology often used in machine learning literature, we refer to this search as a *hyperparameter search*. To identify the most performant values for these hyperparameters, we uniformly sampled 108 distinct parameter sets from this subspace. We then set the hyperparameters for each layer in our pipeline to those of each

Table 2. Hyperparameter search.

Filter	Parameter	Parameter value			
		Search range	Set 20 max AR	Set 75 best mix FR and TDR	Set 88 max AP
Mean	Standard deviation	(2.5, 3.0, 3.5)	2.5	3.5	3.5
Turbine	Frequency range (Hz)	(1.5, 3.5)	(1.5, 3.5)	(1.5, 3.5)	(1.5, 3.5)
	Blur	(11, 13)	13	11	11
Denoise	Blur	(5, 7)	5	5	7
Intensity	Threshold	(100)	100	100	100
Detection	Minimum box (pixels)	(15)	15	15	15
	Maximum box (pixels)	(14335)	14,335	14,335	14,335
Association	Window length (frames)	(3, 4, 5)	3	4	5
	Threshold	(0.06, 0.08, 0.10)	0.10	0.06	0.08

parameter set and evaluated each of the 108 models on the training data to determine those that were most performant. We used the evaluation metrics above to measure each model's performance. From the results of this search on the training set, we chose the three strongest parameter sets to then test on the testing data: that which maximized AP (set 88), that which maximized AR (set 20), and that which satisfied our goal criterion (> 20% FR and > 90% TDR) and maximized MFTDR (set 75). The ranges of this search and the final, most performant values are found in Table 2.

Data split analysis

After applying our model to the training and testing data, we noticed shifts in the performance of our models across each. To compare these two splits and associate these performance discrepancies with other corresponding dataset statistics, we calculated the empirical cumulative distribution function (CDF) of targets by bounding box area (a proxy for target size) for our data. The CDF for a given random variable illustrates the relative probability contribution of each value to the total distribution. We used it to compare the distribution of target size for all targets to that of targets correctly identified by our method, as shown in Fig. 5. To verify that the deviation between these two distributions was significant, we calculated both the Kolmogorov–Smirnov (KS) and the Anderson–Darling test statistics (Anderson and Darling 1954; Smirnov 1939). These provide significance values that quantify the probability of the null hypothesis that one distribution was sampled from the other, that is, that the two come from the same distribution. Kolmogorov–Smirnov does so by comparing the point of greatest difference between the distributions, while Anderson–Darling considers differences across the entire distributions.

Ablation study

To assess the individual contribution of each filtering stage to the final performance of our pipeline, we performed an *ablation study*. We evaluated our method once with each stage of the automated detection pipeline removed. By doing so, we were able to estimate the performance contribution of each stage. This was performed for the three chosen parameter sets (20, 75, and 88) on the testing data.

Assessment

The training and testing performance results on the evaluation metrics for the most-performant hyperparameter sets (20, 75, 88) on the training data can be found in Table 3. While the performance varied somewhat between the training and testing data, the performance of our model exceeded our stated goals: > 90% target detection and > 20% frame removal.

Between the training and testing data precision dropped and recall increased across all hyperparameter sets. In turn, TDR increased while frame removal rates decreased slightly. Putting these factors together, while sets 20 and 75 achieved more desirable performance on the train set, sets 75 and 88 achieved more desirable performance on the test set.

Size vs. performance distributions and statistical tests

In seeking to explain the performance differences between the train and test sets, we identified the sensitivity of our method to target size (Table 1 and Fig. 3). Between the training and testing data, while the mean target size is similar, the median and middle two quartiles of the test set are larger than those of the train set—indicating that there were more, larger targets in the test set. To assess the performance of our methods across target sizes, we compared the CDF of all target sizes to the target sizes of TP detections for both the train and test datasets (Fig. 5). Note that the CDF of TP detections is shifted toward larger targets than the CDF of all targets, indicating that TP detection is skewed toward larger targets. We assessed this shift using the Anderson–Darling and Kolmogorov–Smirnov statistical tests (Fig. 5) and showed that the significance statistics reject the null hypothesis of the two datasets being sampled from the same distribution; that is, that the CDF of all target sizes and those that were correctly detected are significantly divergent. These results indicate two

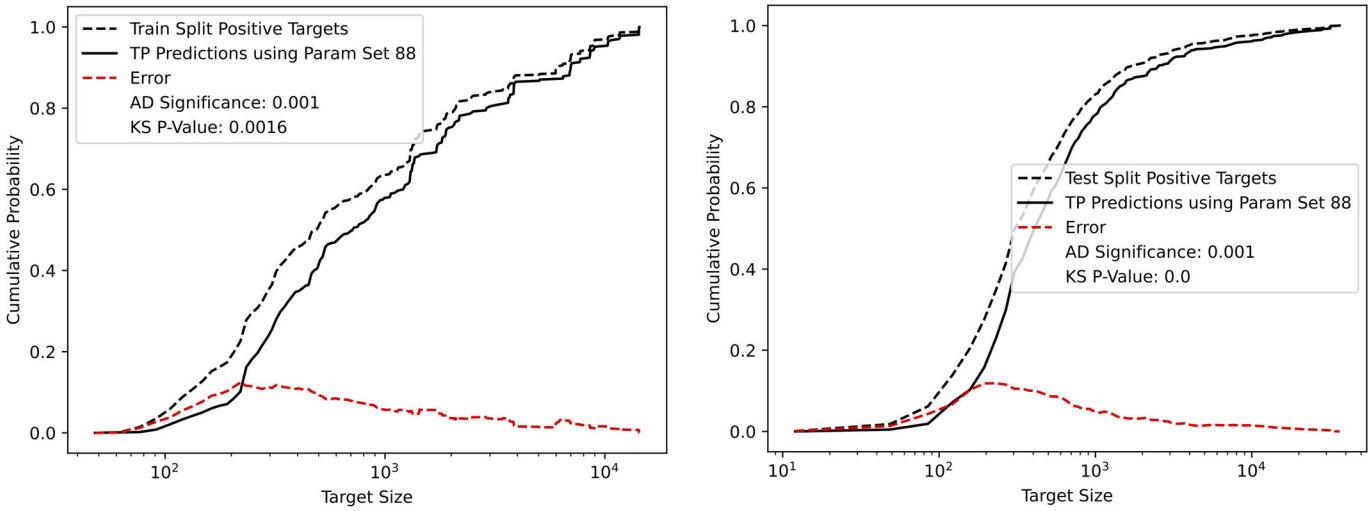

Fig. 5. The figures contain the CDF of targets, the CDF of TP detections, the error between these two CDFs, and the significance in the deviation between these two distributions as measured by the Anderson–Darling and Kolmogorov–Smirnov statistical tests for parameter set 88 on both the training (left) and testing (right) data. Because the CDF of TP predictions is shifted right of the CDF of all targets, we know that correct target prediction is biased toward larger targets. The Anderson–Darling and Kolmogorov–Smirnov tests show that for these models that had large error, the shifts in the distribution because of this size-based detection bias were statistically significant.

Table 3. Evaluation metrics for the three chosen parameter sets from hyperparameter search for the training and testing data. Bold indicates the highest scoring result.

Evaluation metrics	AR		AP		F1		FR		TDR		MFRTDR	
Parameter set	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Set 20	**0.94**	**1.00**	0.19	0.01	0.32	0.02	0.22	0.05	**0.99**	**1.00**	0.61	0.53
Set 75	0.78	0.82	0.28	**0.02**	0.41	0.03	0.56	0.45	0.92	**1.00**	**0.74**	0.73
Set 88	0.63	0.62	**0.34**	**0.02**	**0.44**	**0.04**	**0.70**	**0.70**	0.74	0.98	0.72	**0.84**

things. Firstly, that our method is more performant at detecting larger targets and that this bias in performance is significant. Secondly, that this behavior is at least in part responsible for the divergent performance between the train and test sets.

Ablation study

The results of the ablation study are in Table 4. While frame removal and AP plummeted after removing the denoising filter, background filter, or intensity filter, little change in performance occurred when removing the detection association or turbine filter stages, indicating that they are somewhat redundant steps on this data.

Discussion

In this work, we developed a novel, open-access dataset[1] of labeled targets around a tidal turbine in video rendered from the acoustic camera data captured by a SMC DIDSON. To automatically detect targets around moving turbines and evaluate the efficacy of our method, we introduced an open-source, automated detection pipeline[1] for detecting targets in video and analyzed the results of our method using the aforementioned metrics. By splitting our pipeline into filtering and detection stages, we separate and distinguish between noise filtering (such as turbine backscatter removal) and target detection (applicable to any acoustic camera target detection pipeline). We analyzed the performance of our detection pipeline and the adequacy of our dataset, and in doing so have identified opportunities for improvements and future work. In addition, and through the execution of this work, we have encountered areas of improvement relevant to the collection, labeling, sharing, and processing of data in this domain, which we will discuss in the remainder of this paper.

The PNNL TUNAMELT dataset

A contribution of this work was to develop and openly release a labeled acoustic camera video dataset for target detection. To the best of our knowledge, this is the first around underwater marine energy turbines (Fernandez Garcia et al. 2023; Kay et al. 2022; McCann et al. 2018). Given the novelty

Table 4. Results of the ablation study. Detection association and turbine filtering had little effect on the evaluation results on the test set, indicating that they could readily be removed while maintaining model performance.

Evaluation metrics	AR	AP	F1	FR	TDR	MFRTDR
			All stages			
Set 20	**1.00**	0.01	0.02	0.05	**1.00**	0.53
Set 75	0.82	**0.02**	0.03	0.45	**1.00**	0.73
Set 88	0.62	**0.02**	**0.04**	**0.70**	0.98	**0.84**
			No detection association			
Set 20	**1.00**	0.01	0.02	0.05	**1.00**	0.53
Set 75	0.82	**0.02**	0.03	0.45	**1.00**	0.73
Set 88	0.62	**0.02**	**0.04**	**0.70**	0.98	**0.84**
			No intensity filter			
Set 20	**1.00**	**0.01**	**0.02**	0.00 (−0.05)	**1.00**	0.50 (−0.03)
Set 75	**1.00 (+0.18)**	**0.01 (−0.01)**	**0.02 (−0.01)**	0.01 (−0.44)	**1.00**	0.51 (−0.22)
Set 88	0.98 (+0.36)	**0.01 (−0.01)**	**0.02 (−0.02)**	**0.03 (−0.67)**	**1.00 (−0.02)**	**0.52 (−0.32)**
			No denoising filter			
Set 20	**1.00**	**0.01**	**0.02**	0.00 (−0.05)	**1.00**	0.50 (−0.03)
Set 75	0.99 (+0.17)	**0.01 (−0.01)**	**0.02 (−0.01)**	**0.05 (−0.40)**	**1.00**	**0.53 (−0.20)**
Set 88	0.99 (+0.37)	**0.01 (−0.01)**	**0.02 (−0.02)**	**0.05 (−0.65)**	**1.00 (−0.02)**	**0.53 (−0.31)**
			No turbine filter			
Set 20	**1.00**	0.01	0.02	0.04 (−0.01)	**1.00**	0.52 (−0.01)
Set 75	0.82	**0.02**	0.03	0.45	**1.00**	0.73
Set 88	0.62	**0.02**	**0.04**	**0.70**	0.98	**0.84**
			No background filter			
Set 20	**1.00**	**0.01**	**0.02**	0.00	**1.00**	0.50 (−0.03)
Set 75	**1.00 (+0.18)**	**0.01 (−0.01)**	**0.02 (−0.01)**	0.00 (−0.45)	**1.00**	0.50 (−0.23)
Set 88	**1.00 (+0.38)**	**0.01 (−0.01)**	**0.02 (−0.02)**	**0.09 (−0.69)**	**1.00 (−0.02)**	**0.55 (−0.29)**

Bold indicates the highest scoring result. Values in parenthesis signify the difference in result between the ablated pipeline and the pipeline with all stages.

and nascency of this dataset, there is room for refinement in future work. The dataset statistics in Table 1 and Fig. 3 alongside the target size performance analysis in Fig. 5 highlight some of these possible improvements.

The training and testing data subsets were created to capture many target interactions around turbines (a typically rare occurrence). However, in doing so, the distribution of performance-affecting video characteristics—such as target size—became biased between the two datasets, making the training data less representative of the testing data, and thus a weaker surrogate for model development. A means of improving this would be to redefine the training and testing data of the dataset to ensure that the two are drawn from similar distributions of performance-affecting characteristics (such as target size, target presence, turbine presence, or others).

Evaluation metrics

In this work, we used both standard and atypical metrics for evaluating our approaches. Precision, recall, and *F*1-score are standard metrics for evaluating classification tasks. Frame removal percentage, TDR, and the mean of these two are not standard. We chose to use, and often focused our analysis on, these atypical metrics because of our focus on providing information for a human reviewer, that is, a reduced dataset for review. Precision, recall, and *F*1-score each assess a model's ability to detect a target *in every frame*, while TDR assesses whether the model would notify an analyst to at least one frame in which each target appeared (after which the analyst could observe the target across neighboring frames). For our targeted application, we felt that these atypical metrics better assessed our progress toward that goal.

While we performed object detection, we did not use common-object detection metrics in the evaluation of our methods, but rather binary classification metrics. Our focus on informing an analyst is again responsible for this choice. Rather than seeking to accurately track a target's location in a frame, the goal of our work was to alert an analyst to whenever a target was present. As such, we sought to predict in which frames a target was present, but we did so by detecting the box around plausible targets, then converting the boxes to per-frame binary classification predictions. As progress is made in this space, the precision of predictions or their locations may become the focus of future work. Hence, as this space evolves, a greater focus on precision and recall, or the

inclusion of common detection metrics such as intersection over union or multiple object tracking accuracy, may be necessitated (Bernardin and Stiefelhagen 2008; Leal-Taixé et al. 2017).

Choosing between deep learning and classical computer vision-based approaches

Although we developed a dataset suitable for machine learning development, in this work we chose to use classical computer vision approaches to first filter frames before performing target detection with a non-deep-learning-based approach. Across other problem spaces of computer vision, deep learning has demonstrated state-of-the-art performance. Therefore, the critical reader may question why we did not do so in this work. Our rationale is as follows. Given the size of our dataset, a deep learning model first pretrained on a larger, more ubiquitous dataset would likely have been necessary. However, the transfer of a pretrained model from one dataset to another assumes the existence of common features between the two. At the time of writing, no such large-scale datasets exist for acoustic camera video. Given the low resolution, noise, absence of color features, dissimilarity between our targets and typical target classes, general dissimilarity between underwater acoustic camera video and red, green, blue (RGB) video of common objects, and the results of prior works using transfer learning in this domain, the authors suspected that any transfer learning between a large-scale RGB dataset would not meet our performance requirements (> 90% target detection and > 20% frame removal). As previously discussed, we also saw value in distinguishing and developing the filtering and detection steps separately. Provided our filtering methods do not remove information useful to the detector, they neither preclude nor diminish the use of deep learning-based detectors in future work. Rather, filtering methods make the task of detecting objects in any manner easier, and in this way, we sought to maximize performance with a simple, out-of-the-box detector (Bradski 2000; Suzuki 1985) as a baseline approach with the expectation that future work would readily substitute it for a more powerful detection approach.

The performance of our approach

Using our automated detection pipeline, we sought to detect 90% of targets while removing 20% of empty frames in an effort to assist an analyst. On the test set, our best model detected 98% of targets while removing 70% of the video frames (Table 2). However, in designing a method that was sensitive enough to detect nearly all targets, our method had low precision. On the test set, only 2% of detections would be TPs, while on the train set, 34% were TPs (this was due to the difference in target sparsity between the two sets). While reviewing this subset of detected frames is likely faster than reviewing entire videos, the cognitive load of jumping between frames using nonsequential detections may make doing so more burdensome to the reviewer. Hence, despite the TDR and frame removal rate, future work will be needed to improve the precision of these methods to provide a stronger aid for human review.

The utility of the turbine filter stage

While the turbine filter sought to remove oscillatory, high-intensity pixels from the video, through the ablation study it became clear that the background removal filter had already done so. Upon visual inspection of the binary masks and the intermediate video outputs of our pipeline in Fig. 4, it is clear that while the turbine filter removes some remaining oscillatory segments of the image, the motion of the turbine is slow and repetitive enough that it is regarded as sufficiently persistent to be considered static by our background filter. This is likely due to the long time window on which the background-removal filter is being calculated in its current, offline implementation. In this work, we are using the entire length of the video snippet as the window for this background removal. As such, relative to the length of these videos, the roughly 1 Hz oscillation of the turbine causes the turbine blade to reappear often enough in the same location as to be characterized as background. Because this will not be the case in real-time implementations that seek to filter static background using a moving, windowed-average filter, we hypothesize that such future work will need to implement a separate sliding DFT or other periodic filter to remove the oscillation of the turbine (Bradford et al. 2005).

The utility of the detection association stage

Our detection association algorithm had little effect on the performance of our method (Table 4). This is likely due in part to the relative simplicity of our approach. Our algorithm for detection association only included two similarity metrics—a size metric and a distance metric. Furthermore, the threshold that prevents the association of dissimilar detections is applied against the summation of these two costs—resulting in potential mis-associations of detections (e.g., a detection whose centroid is in roughly the same location, but changes shape unrealistically might still be associated). Future work may consider including a velocity metric (such as the Kalman Filter), path constraints, or an association algorithm (such as Kuhn–Munkres) that finds the global optimum association to improve performance (Aharon et al. 2022). These may help improve the precision of the detection association, and in doing so, the precision of the entire pipeline.

Labeled, open-access datasets

Acoustic cameras are important tools for monitoring fish interactions with turbines where water turbidity precludes the use of optical sensors. However, as with most environmental monitoring, the large amount of accumulated data without targets present is time-consuming and expensive to review and process. Common, open-access, labeled datasets would allow developers to:

- Reduce costs by sharing field-collected data in a common, readily usable format.
- Prototype and test approaches rapidly and in a standardized fashion.
- Evaluate the performance of a method against ground-truth labels.
- Compare approaches using common metrics on shared data.
- Make available the development of detection approaches to more researchers.

Like datasets in the computer vision community for common-object classification, detection, semantic-segmentation, and other classes of problems, we hope that by cultivating an ecosystem of open-source data and code that further progress can be made in automating acoustic data processing (Deng et al. 2009; Lin et al. 2014).

Future data collections and dataset creation efforts

The advances made by deep learning in other domains come from the prevalence of data-capturing technologies: social media, internet of things, and enhanced internet connectivity. To translate such advancements to the marine energy domain, similar volumes and diversities of data need to be generated.

Labeling is often the most burdensome step in the creation of such large computer vision datasets suitable for machine learning. In this work, we used the open-source, semi-automated labeling tool—CVAT—to expedite this process. Doing so allowed us to linearly interpolate target bounding boxes between start and end positions without manually labeling every video frame. Automations such as this, automatic object segmentation, or even automated labeling with other detection models can greatly increase the quality and speed of creating labeled datasets. In this vein, our automated detection pipeline could also aid in the creation of future datasets. By removing frames without objects of interest and providing bounding box detections in a context-agnostic manner, it could be used to reduce the burden associated with acoustic camera dataset annotation, as it might for acoustic camera video review.

Future work seeking to make additional, labeled datasets should consider collecting and labeling data from other sites, using modern, higher-resolution sensors (e.g., SMC ARIS), and including as metadata or corresponding data multiple views of acoustic camera video and the environmental and sensing conditions of the collection site. Such augmentations could also include sensor pose in global coordinates, 3D meshes for static objects in each scene (Cordts et al. 2016; Liao et al. 2023; Reizenstein et al. 2021), locations of the ground plane and water surface relative to the sensor, time-aligned water turbidity measurements, and calibrated multi-viewpoint data from either additional acoustic cameras or additional sensing modalities (Macenski et al. 2022; MRD 2022; Oettershagen et al. 2015). By including such augmentations, high-resolution semantic maps can begin to be created and used to supervise more modern and complex approaches for target detection, site characterization, and site modeling, and thus better inform regulator understanding of sites and the associated collision risks (Cheng et al. 2022; Liang et al. 2019; Maturana et al. 2018). These methods would furthermore contribute to improved turbine maintenance through digital twinning. Simultaneously, community standards for data formats and sharing need to be established to enable data sharing and rapid algorithm development. Finally, these datasets should be collocated, made open-access, and popularized via workshops, competitions, and benchmarks publicized at top-tier conferences (Kristan et al. 2016; Leal-Taixé et al. 2015).

Comments and recommendations

- Future data collections should be labeled and evaluated using standardized formats, shared openly alongside detection approaches, collected, and evaluated via benchmarks, and publicized at conferences to promote development in this space. Our work presents a possible approach for doing so that may serve as a model for the collision risk community.
- This work developed a novel, labeled, open-access dataset and a performant, open-access automated target detection approach that can be used to reduce the costs of marine energy site characterization and monitoring.
- Data quality, sensor fidelity, filtering, and the detection approach all equally contribute to final target detection success. By splitting our approach into stages and openly sharing our approach and metadata, we provide an extensible and transparent baseline approach for developing successful pipelines for this domain.
- Detecting small targets with single-view acoustic camera imagery is challenging due to difficult backscatter characteristics at high incident angles on small target areas and due to low frame rates. Persistent, multi-sensor, multi-viewpoint deployments with corresponding environmental data bundled into common, open data formats can be used to effectively monitor and characterize marine energy sites and reduce project costs and timelines.

Author Contributions

Theodore Nowak: conceptualization (equal), data curation (equal), formal analysis (lead), investigation (lead), methodology (lead), resources (lead), software (lead), supervision (equal), validation (lead), visualization (lead), writing – original draft preparation (lead), writing – review and editing (lead). Garrett Staines: conceptualization (equal), data curation (equal), funding acquisition (lead), project administration (lead), supervision (equal), writing – original draft preparation (contribution), writing – review and editing (contribution). Blerim Abdullai: formal analysis (contribution), investigation

(contribution), software (secondary), writing – review and editing (contribution).

Acknowledgments

The authors thank Eddie Pablo for labeling data and Susan Ennor, Gayle Zydlewski, Haley Viehman, Emma Cotter, and Noriaki Kono for improvements to earlier versions of this manuscript, as well as Ocean Renewable Power Company's collaboration and support. This research was funded by the US Department of Energy Water Power Technologies Office under contract DE-AC05-76RL01830 with the Pacific Northwest National Laboratory.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data and code contributed by this work are openly available at https://github.com/tsnowak/pnnl-tunamelt. The data have also been uploaded to MHKDR at https://mhkdr.openei.org/submissions/633.

References

Aharon, N., R. Orfaig, and B.-Z. Bobrovsky. 2022. "Bot-Sort: Robust Associations Multi-Pedestrian Tracking." arXiv Preprint arXiv:2206.14651.

Anderson, T. W., and D. A. Darling. 1954. "A Test of Goodness of Fit." *Journal of the American Statistical Association* 49: 765–769. https://doi.org/10.1080/01621459.1954.10501232.

ARISfish. 2020. "Sound Metrics Corporation." http://www.soundmetrics.com/.

Baumgartner, L. J., N. Reynoldson, and D. M. Gilligan. 2006. "Mortality of Larval Murray Cod (Maccullochella Peelii Peelii) and Golden Perch (*Macquaria ambigua*) Associated with Passage through Two Types of Low-Head Weirs." *Marine and Freshwater Research* 57: 187–191. https://doi.org/10.1071/MF05098.

Bernardin, K., and R. Stiefelhagen. 2008. "Evaluating Multiple Object Tracking Performance: The Clear Mot Metrics." *EURASIP Journal on Image and Video Processing* 2008: 1–10. https://doi.org/10.1155/2008/246309.

Bevelhimer, M., C. Scherelis, J. Colby, and M. A. Adonizio. 2017. "Hydroacoustic Assessment of Behavioral Responses by Fish Passing Near an Operating Tidal Turbine in the East River, New York." *Transactions of the American Fisheries Society* 146: 1028–1042. https://doi.org/10.1080/00028487.2017.1339637.

Bradford, R., D. Richard, and J. Fitch. 2005. "Sliding is Smoother Than Jumping." In International Computer Music Conference, 287–290. International Computer Music Association (ICMA).

Bradski, G. 2000. "The OpenCV Library." Dr. Dobb's Journal of Software Tools.

Braga, L. T. M. D., A. Giraldo, and A. L. Godinho. 2022. "Evaluation of Three Methods for Manually Counting Fish in Dam Turbines Using DIDSON." *Hydrobiologia* 849: 309–321. https://doi.org/10.1007/s10750-021-04605-x.

Burwen, D. L., S. J. Fleischman, and J. D. Miller. 2010. "Accuracy and Precision of Salmon Length Estimates Taken from DIDSON Sonar Images." *Transactions of the American Fisheries Society* 139: 1306–1314. https://doi.org/10.1577/T09-173.1.

Capoccioni, F., C. Leone, D. Pulcini, M. Cecchetti, A. Rossi, and E. Ciccotti. 2019. "Fish Movements and Schooling Behavior Across the Tidal Channel in a Mediterranean Coastal Lagoon: An Automated Approach Using Acoustic Imaging." *Fisheries Research* 219: 105318. https://doi.org/10.1016/j.fishres.2019.105318.

Cheng, Q., N. Zeller, and D. Cremers. 2022. "Vision-Based Large-Scale 3D Semantic Mapping for Autonomous Driving Applications." In 2022 International Conference on Robotics and Automation (ICRA), 9235–9242. IEEE.

Copping, A. E., and L. G. Hemery. 2020. OES-Environmental 2020 State of the Science Report. Pacific Northwest National Lab (PNNL).

Cordts, M., M. Omran, S. Ramos, et al. 2016. "The Cityscapes Dataset for Semantic Urban Scene Understanding." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3213–3223. IEEE.

Cotter, E., and G. Staines. 2023. "Observing Fish Interactions With Marine Energy Turbines Using Acoustic Cameras." *Fish and Fisheries* 24: 1020–1033. https://doi.org/10.1111/faf.12782.

CVAT. 2022. "Computer Vision Annotation Tool." https://zenodo.org/records/4009388.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. "Imagenet: A Large-Scale Hierarchical Image Database." In 2009 IEEE Conference on Computer Vision and Pattern Recognition, 248–255. Ieee.

Egg, L., J. Pander, M. Mueller, and J. Geist. 2018. "Comparison of Sonar-, Camera- and Net-Based Methods in Detecting Riverine Fish-Movement Patterns." *Marine and Freshwater Research* 69: 1905–1912. https://doi.org/10.1071/MF18068.

Eggleston, M. R., S. W. Milne, M. Ramsay, and K. P. Kowalski. 2020. "Improved Fish Counting Method Accurately Quantifies High-Density Fish Movement in Dual-Frequency Identification Sonar Data Files From a Coastal Wetland Environment." *North American Journal of Fisheries Management* 40: 883–892. https://doi.org/10.1002/nafm.10451.

Fernandez Garcia, G., T. Corpetti, M. Nevoux, L. Beaulaton, and F. Martignac. 2023. "AcousticIA, a Deep Neural Network for Multi-Species Fish Detection Using Multiple Models of Acoustic Cameras." *Aquatic Ecology* 57: 881–893. https://doi.org/10.1007/s10452-023-10004-2.

Gillespie, D., G. Hastie, J. Montabaranom, et al. 2023. "Automated Detection and Tracking of Marine Mammals in the Vicinity of Tidal Turbines Using Multibeam Sonar." *Journal of Marine Science and Engineering* 11: 2095. https://doi.org/10.3390/jmse11112095.

Grote, A. B., M. M. Bailey, J. D. Zydlewski, and J. E. Hightower. 2014. "Multibeam Sonar (DIDSON) Assessment of American Shad (*Alosa sapidissima*) Approaching a Hydroelectric Dam." *Canadian Journal of Fisheries and Aquatic Sciences* 71: 545–558. https://doi.org/10.1139/cjfas-2013-0308.

Gurney, W., L. O. Brennan, P. J. Bacon, et al. 2014. "Objectively Assigning Species and Ages to Salmonid Length Data from Dual-Frequency Identification Sonar." *Transactions of the American Fisheries Society* 143: 573–585. https://doi.org/10.1080/00028487.2013.862185.

Han, J., N. Honda, A. Asada, and K. Shibata. 2009. "Automated Acoustic Method for Counting and Sizing Farmed Fish During Transfer Using DIDSON." *Fisheries Science* 75: 1359–1367. https://doi.org/10.1007/s12562-009-0162-5.

Hasselman, D. J., D. R. Barclay, R. Cavagnaro, et al. 2020. "Chapter 10: Environmental Monitoring Technologies and Techniques for Detecting Interactions of Marine Animals with Turbines." In 2020 State of the Science Report, edited by A. E. Copping and L. G. Hemery, 176–213. Pacific Northwest National Laboratory.

Helminen, J., and T. Linnansaari. 2021. "Object and Behavior Differentiation for Improved Automated Counts of Migrating River Fish Using Imaging Sonar Data." *Fisheries Research* 237: 105883. https://doi.org/10.1016/j.fishres.2021.105883.

Hightower, J. E., K. J. Magowan, L. M. Brown, and D. A. Fox. 2013. "Reliability of Fish Size Estimates Obtained From Multibeam Imaging Sonar." *Journal of Fish and Wildlife Management* 4: 86–96. https://doi.org/10.3996/102011-JFWM-061.

Holmes, J. A., G. M. Cronkite, H. J. Enzenhofer, and T. J. Mulligan. 2006. "Accuracy and Precision of Fish-Count Data From a 'Dual-Frequency Identification Sonar' (DIDSON) Imaging System." *ICES Journal of Marine Science* 63: 543–555. https://doi.org/10.1016/j.icesjms.2005.08.015.

Kay, J., P. Kulits, S. Stathatos, et al. 2022. "The Caltech Fish Counting Dataset: A Benchmark for Multiple-Object Tracking and Counting." *Lecture Notes in Computer Science* 13668: 290–311. https://doi.org/10.1007/978-3-031-20074-8_17.

Key, B. H., J. D. Miller, S. J. Fleischman, and J. Huang. 2016. Chinook Salmon Passage in the Kenai River at River Mile 13.7 Using Adaptive Resolution Imaging Sonar, 2015. Alaska Department of Fish and Game.

Kilcher, L., M. Fogarty, and M. Lawson. 2021. Marine Energy in the United States: An Overview of Opportunities. National Renewable Energy Laboratory (NREL).

Kristan, M., J. Matas, A. Leonardis, et al. 2016. "A Novel Performance Evaluation Methodology for Single-Target Trackers." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38: 2137–2155. https://doi.org/10.1109/TPAMI.2016.2516982.

Kuhn, H. W. 1955. "The Hungarian Method for the Assignment Problem." *Naval Research Logistics Quarterly* 2: 83–97. https://doi.org/10.1002/nav.3800020109.

Leal-Taixé, L., A. Milan, I. Reid, S. Roth, and K. Schindler. 2015. "Motchallenge 2015: Towards a Benchmark for Multi-Target Tracking." arXiv Preprint arXiv:1504.01942.

Leal-Taixé, L., A. Milan, K. Schindler, D. Cremers, I. Reid, and S. Roth. 2017. "Tracking the Trackers: An Analysis of the State of the Art in Multiple Object Tracking." arXiv Preprint arXiv:1704.02781.

Liang, M., B. Yang, Y. Chen, R. Hu, and R. Urtasun. 2019. "Multi-Task Multi-Sensor Fusion for 3d Object Detection." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 7345–7353. IEEE.

Liao, Y., J. Xie, and A. Geiger. 2023. "KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2d and 3d." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45: 3292–3310. https://doi.org/10.1109/TPAMI.2022.3179507.

Lin, T.-Y., M. Maire, and S. Belongie. 2014. "Microsoft Coco: Common Objects in Context." In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13, 740–755. Springer.

Macenski, S., T. Foote, B. Gerkey, C. Lalancette, and W. Woodall. 2022. "Robot Operating System 2: Design, Architecture, and Uses in the Wild." *Science Robotics* 7: eabm6074. https://doi.org/10.1126/scirobotics.abm6074.

Maturana, D., P.-W. Chou, M. Uenoyama, and S. Scherer. 2018. "Real-Time Semantic Mapping for Autonomous off-Road Navigation." In Field and Service Robotics: Results of the 11th International Conference, 335–350. Springer.

Matzner, S., C. K. Trostle, G. J. Staines, R. E. Hull, A. Avila, and G. E. Harker-Klimes. 2017. Triton: Igiugig Video Analysis-Project Report. Pacific Northwest National Laboratory.

McCann, E., L. L. Li, K. Pangle, N. Johnson, and J. Eickholt. 2018. "An Underwater Observation Dataset for Fish Classification and Fishery Assessment." *Scientific Data* 5: 180190. https://doi.org/10.1038/sdata.2018.190.

Melvin, G. D., and N. A. Cochrane. 2015. "Multibeam Acoustic Detection of Fish and Water Column Targets at High-Flow Sites." *Estuaries and Coasts* 38: 227–240. https://doi.org/10.1007/s12237-014-9828-z.

MRD. 2022. Marine Robotics Dataset. Australian Centre for Field Robotics.

Oettershagen, P., T. Stastny, T. A. Mantel, et al. 2015. "Long-Endurance Sensing and Mapping using a Hand-Launchable Solar-Powered UAV." In International Symposium on Field and Service Robotics. International Foundation of Robotics Research.

Rakowitz, G., M. Tušer, M. Říha, T. Jůza, H. Balk, and J. Kubečka. 2012. "Use of High-Frequency Imaging Sonar (DIDSON) to Observe Fish Behaviour Towards a Surface

Trawl." *Fisheries Research* 123: 37–48. https://doi.org/10.1016/j.fishres.2011.11.018.

Reizenstein, J., R. Shapovalov, P. Henzler, L. Sbordone, P. Labatut, and D. Novotny. 2021. "Common Objects in 3d: Large-Scale Learning and Evaluation of Real-Life 3d Category Reconstruction." In Proceedings of the IEEE/CVF International Conference on Computer Vision, 10901–10911. IEEE.

Smirnov, N. V. 1939. "On the Estimation of the Discrepancy between Empirical Curves of Distribution for Two Independent Samples." *Bulletin de la Société Mathématique de l'Université de Moscou* 2: 3–14.

Smit, M., E. Winter, and P. Scheijgrond. 2016. Tidal Energy Fish Impact: Method Development to Determine the Impact of Open Water Tidal Energy Converters on Fish. Netherlands Institute for Sea Research (NIOZ).

Staines, G., G. Zydlewski, and H. Viehman. 2019. "Changes in Relative Fish Density Around a Deployed Tidal Turbine During On-Water Activities." *Sustainability* 11: 6262. https://doi.org/10.3390/su11226262.

Suzuki, S. 1985. "Topological Structural Analysis of Digitized Binary Images by Border Following." *Computer Vision, Graphics, and Image Processing* 30: 32–46. https://doi.org/10.1016/0734-189X(85)90016-7.

Tušer, M., J. Frouzová, H. Balk, M. Muška, T. Mrkvička, and J. Kubečka. 2014. "Evaluation of Potential Bias in Observing Fish with a DIDSON Acoustic Camera." *Fisheries Research* 155: 114–121. https://doi.org/10.1016/j.fishres.2014.02.031.

van Keeken, O. A., R. van Hal, H. Volken Winter, I. Tulp, and A. B. Griffioen. 2020. "Behavioural Responses of Eel (Anguilla Anguilla) Approaching a Large Pumping Station with Trash Rack Using an Acoustic Camera (DIDSON)." *Fisheries Management and Ecology* 27: 464–471. https://doi.org/10.1111/fme.12427.

Viehman, H. A., and G. B. Zydlewski. 2015. "Fish Interactions With a Commercial-Scale Tidal Energy Device in the Natural Environment." *Estuaries and Coasts* 38: 241–252. https://doi.org/10.1007/s12237-014-9767-8.

Williamson, B. J., P. Blondel, L. D. Williamson, and B. E. Scott. 2021. "Application of a Multibeam Echosounder to Document Changes in Animal Movement and Behaviour around a Tidal Turbine Structure." *ICES Journal of Marine Science* 78: 1253–1266. https://doi.org/10.1093/icesjms/fsab017.

Xie, Y., and F. J. Martens. 2014. "An Empirical Approach for Estimating the Precision of Hydroacoustic Fish Counts by Systematic Hourly Sampling." *North American Journal of Fisheries Management* 34: 535–545. https://doi.org/10.1080/02755947.2014.892546.

Submitted 06 June 2025
Revised 21 October 2025
Accepted 02 December 2025