“One Minus” Approach - YOLOv8 Model

**Project Overview**

Wave energy converters ([WEC](https://www.coastalwiki.org/wiki/Wave_energy_converters)) transform energy from ocean waves into electricity. The technology is nascent, with a single operating test site in the US. Regulatory agencies and scientists are interested in the noise created by devices to ensure WECs do not negatively affect marine life. Researchers have collected underwater audio recordings of WECs and their environments, but samples contain many sounds with no effective method of source attribution.

Our team used the underwater audio recordings to generate [spectrograms](https://pnsn.org/spectrograms/what-is-a-spectrogram#:~:text=A%20spectrogram%20is%20a%20visual,energy%20levels%20vary%20over%20time.) (graphs of audio time-frequency composition) and identify WEC sounds. We used two image recognition algorithms, YOLOv8 and VGGish, to accomplish the task. This work was challenging given the complexity of the sonic environment, the unknown composition of WEC sounds, and the subject matter expertise required to annotate data.

**Introduction**

In our project, we utilized the sophisticated object detection capabilities of the Ultralytics YOLOv8 model to classify sounds within spectrograms, with a specific focus on identifying sounds emitted by Wave Energy Converters (WECs). Acknowledging the impracticality of compiling comprehensive training data for all possible representations of WEC sounds, and due to the unique and variable sound profiles generated by differing WEC devices over time, our team devised the “ONE MINUS” method. This dual-model strategy employs two distinct YOLOv8 models: the “ONE” model, which detects all discernible sounds in a spectrogram using a unified "sound" label, and the “MINUS” model, which classifies sounds in the spectrogram with more granular labels.

For each image, we compare the output bounding boxes generated by the “ONE” model against those from the “MINUS” model. By removing the known sounds identified by the “MINUS” model from the “ONE” model’s outputs, we isolate the unidentified sounds for further examination. Ideally, sounds generated by WEC devices would be highlighted in the unidentified sounds for review by a marine acoustics expert.

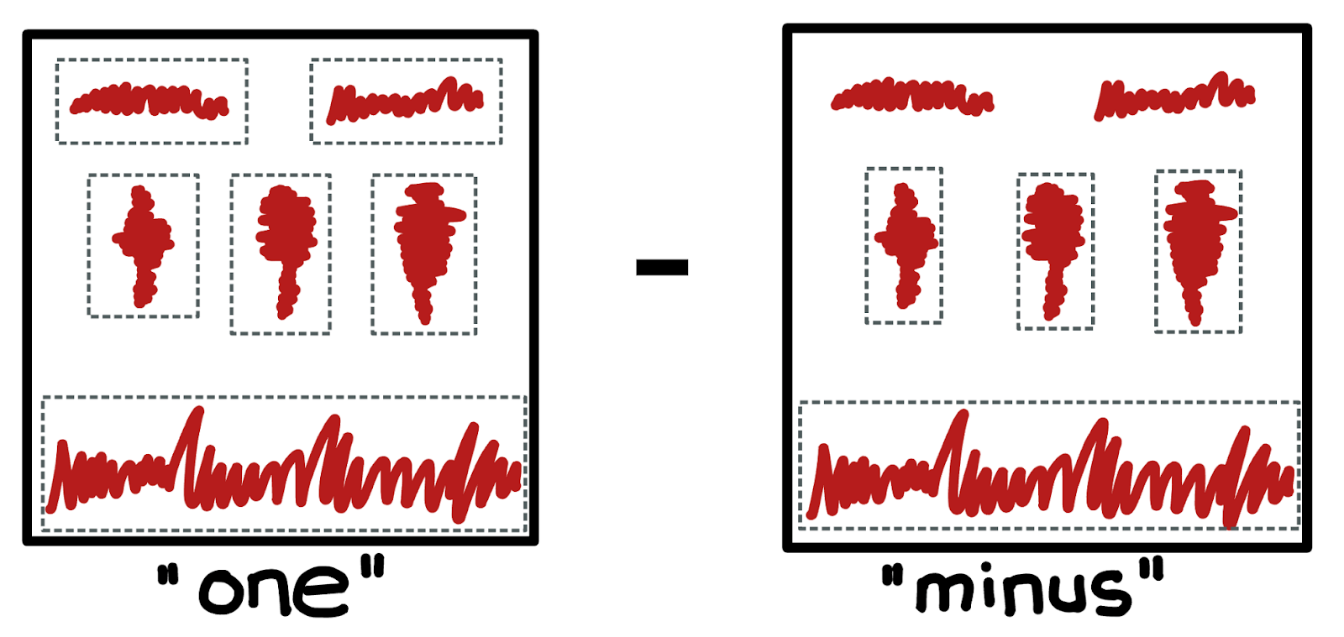


Figure 1: Visual representation of the “One Minus” approach

**Data**

The project sponsor provided raw audio files and spectrograms from two WEC environments.

**Dataset A**

* Recordings from the Fred Olsen WEC dated December 1st, 2018 - February 28th, 2019
* 39,746 spectrograms
* Annotations for ~400 spectrograms with the following classes: humpback, airplane, boat, helicopter, flow noise, mooring

**Dataset B**

* Recordings from the Azura WEC dated January 6th, 2016 - April 12th, 2016
* 42,338 unannotated spectrograms

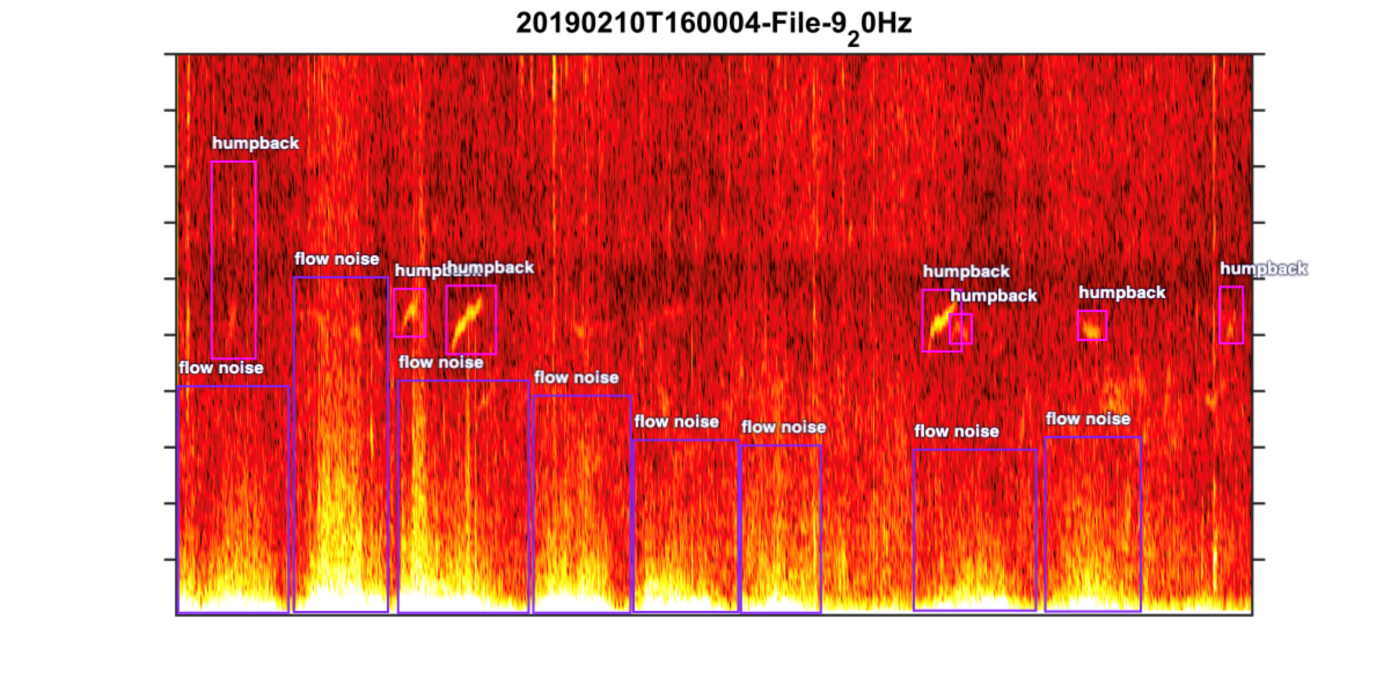


Figure 2: Sample Spectrogram

**Platforms**

We used several platforms throughout our project.

**Annotation: Roboflow**

* Roboflow provided a user-friendly interface for annotating data, allowing for efficient labelling of spectrograms and associated sound classifications used in model training.

**Hyperparameter Tuning Experiments: Azure ML**

* Azure Machine Learning enabled the parallel orchestration of hyperparameter tuning experiments to optimize the performance of the YOLOv8 models.
* We chose Azure over other cloud platforms due to the availability of credits through the MSDS program.
* Users can choose to run YOLOv8 in any number of environments and are not limited to Azure or Google Colaboratory.

**Code Development and Testing: Google Colaboratory**

* Google Colaboratory served as our primary environment for developing and testing code.
* Its integration with Google Drive and free access to GPU resources provided a convenient and cost-effective solution for collaborative coding and experimentation.
* Colaboratory's compatibility with popular Python libraries, such as TensorFlow, facilitated seamless integration with our machine learning workflow.

**Methodology**

**Data Preparation and Annotation**

To fine-tune the Ultralytics YOLOv8 model for our custom dataset, we undertook substantial preparatory work to format image files and their corresponding annotations correctly. Our Project Sponsor supplied annotations for Dataset A in the LabelMe JSON format, necessitating conversion to the YOLOv8-compatible .txt format. We chose to use Roboflow as our annotation tool due to its collaborative capabilities and the native support for importing Dataset A’s LabelMe format.

**Annotation Classes and Conversion**

Using Roboflow, we converted and generated annotations for Dataset A and B respectively in YOLOv8 compatible .txt files for training the "MINUS" and "ONE" model. Dataset A included classes such as Humpback, Flow Noise, Airplane, Helicopter, Boat, and Mooring, while Dataset B added classes like Interesting and WEC.

For the "MINUS" model, we excluded Fish and Mooring classes due to the lack of consistent annotation. For the "ONE" model, we simplified the remaining sound classifications using Roboflow’s dataset versioning feature and aggregated all class labels into a single "sound" label. This process allowed us to keep all annotations in Roboflow, modifying them as needed to create dataset versions tailored for the "MINUS" and "ONE" model, respectively.

**Model Selection and Evaluation**

Our objective was to develop .pt files for both the "ONE" and "MINUS" models, trained with annotated images from Datasets A and B. Initial model evaluation, based on mAP50 scores and runtime comparisons across various YOLOv8 models (yolov8n.pt, yolov8s.pt, yolov8m.pt, yolov8l.pt, yolov8x.pt), led us to choose yolov8m.pt as our foundational model. Performance testing also revealed significant differences in runtime between GPU (approx. 30 minutes for 100 epochs on Google Colab) and CPU (approx. 2.5 hours on Azure ML CPU clusters).

**Hyperparameter Tuning**

Given limited access to GPU compute resources, we utilized Azure ML's compute clusters for extensive parallel hyperparameter tuning, employing a systematic approach to refine our models. The hyperparameter search space, initially broad, was progressively narrowed based on empirical results from our initial tuning phase, optimizing for both the "ONE" and "MINUS" models concurrently. The values highlighted in blue are those selected for the final models.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Epochs | Patience | Batch | Optimizer | Learning rate | IoU | Dropout | Cropped Image |
| 100 | 20 | 8 | Adam | 0.01 | 0.3 | 0.2 | Y |
| 200 | 50 | 16 | SGD | 0.001 | 0.5 | 0.5 | N |
| 500 | 100 | 32 |  |  |  |  |  |

Table 1: Hyperparameter Experiments

**Other Experiments**

We explored two methodologies for model training:

* Sequential Training: Beginning with the yolov8m.pt model, we first trained and tuned using Dataset A, then integrated Dataset B for further refinements.
* Integrated Training: Starting with the yolov8m.pt model, we simultaneously trained on both Datasets A and B, aiming for a unified model optimization from the outset.

**Results**

**“Minus” Model Performance**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Starting .pt | Input Data | Overall mpa50 |
| minus-iter-2 | **8m.pt** | **Fred Olsen** | **0.837** |
| minus-iter-3 | **Best.pt from minus-iter-2** | **Azura** | **0.70** |
| combined | **8m.pt** | **Fred Olsen & Azura** | **0.806** |

Table 2: Best “Minus” iterations

Based on the results in the tables above, the "MINUS" model showed a higher performance when trained specifically on Dataset A, achieving a peak mAP50 score of 0.837. When the model was trained on a combination of Dataset A and Dataset B, there was a marginal decrease in performance from the peak, with a mAP50 score of 0.806. The Confusion Matrix for the results of the “Minus” model is below.

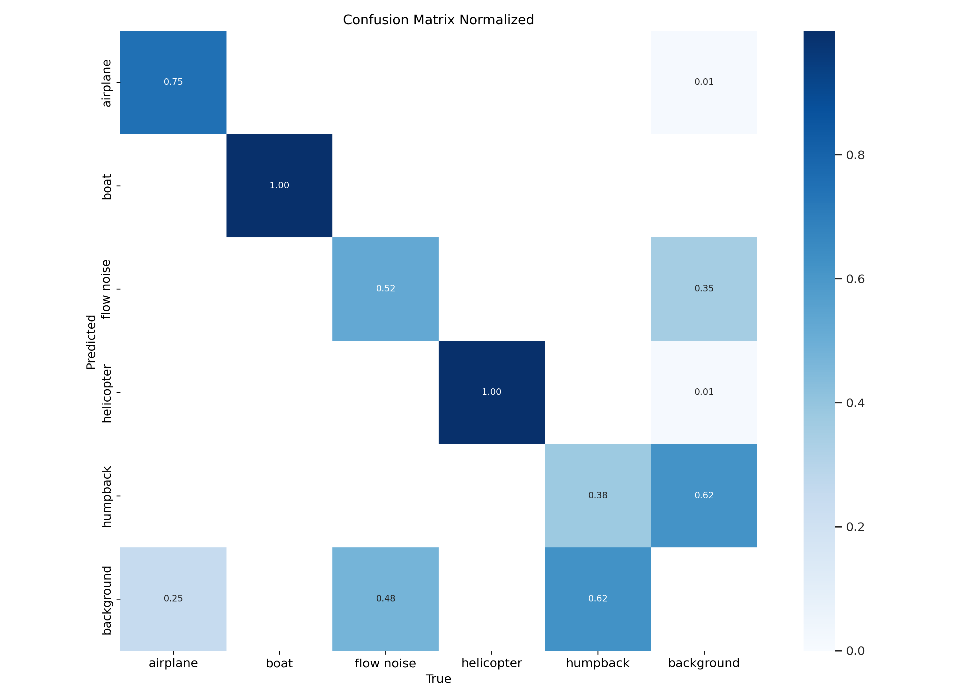


Figure 3: Best “Minus” Model Confusion Matrix

**“One” Model Performance**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Starting .pt | Input Data | Overall mpa50 |
| one-iter-1 | **8m.pt** | **Fred Olsen** | **0.705** |
| one-iter-2 | **Best.pt from one-iter-1** | **Azura** | **0.416** |
| combined | **8m.pt** | **Fred Olsen & Azura** | **0.572** |

Table 3: Best “One” iterations

For the "ONE" model, initial training on Dataset A resulted in a mAP50 of 0.705. The performance substantially decreased when the model was trained with Dataset B, dropping to a mAP50 of 0.416. Combined training on both datasets yielded a mAP50 of 0.572, which is better than the performance on Dataset B alone but not as high as the initial training on Dataset A. The Confusion Matrix for the results of the “One” model is attached below.

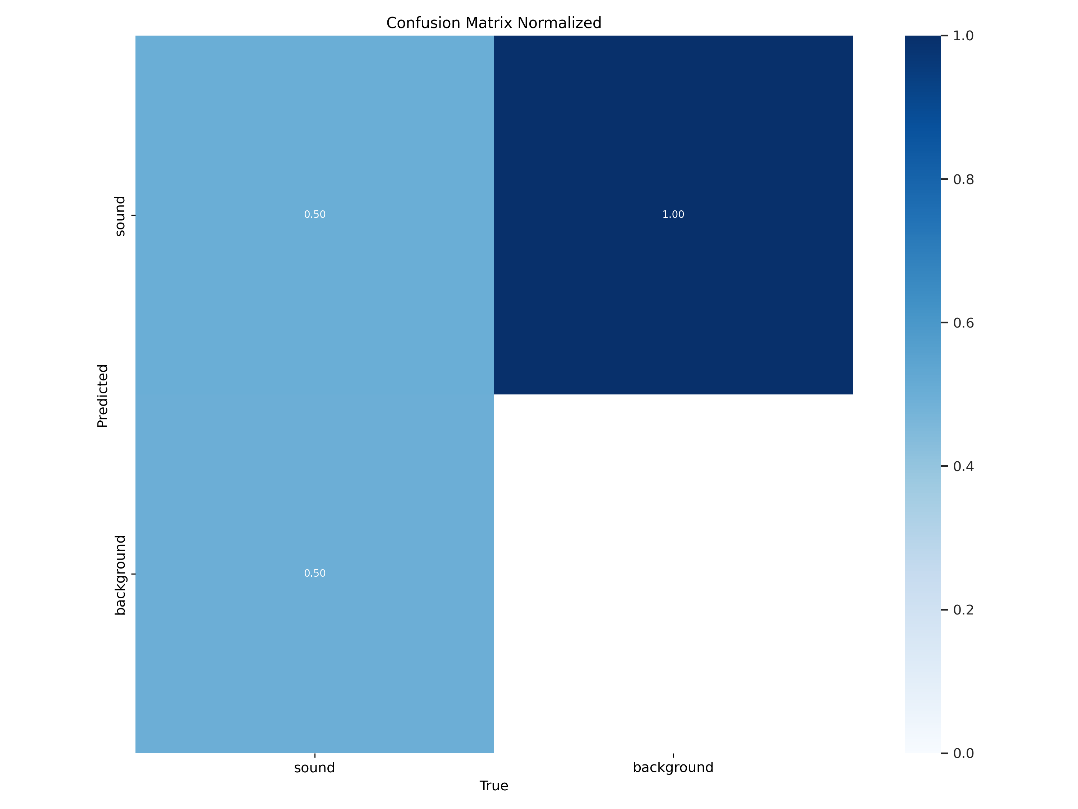


Figure 4: Best “One” Model Confusion Matrix

These results suggest that the "ONE" model did not perform as robustly as the "MINUS" model, especially when Dataset B was included. Moreover, both models showed improved performance when trained exclusively on Dataset A as opposed to the combined datasets. This indicates that Dataset A may have more consistent or clearer features for training the models, or perhaps Dataset B introduces complexity or variability that the models found challenging to generalize when combined with Dataset A.

**Intersection over Union Framework**

The Intersection over Union (IoU) framework serves as a fundamental method for evaluating the performance of object detection models, measuring the degree of overlap between predicted and ground truth bounding boxes. In our study, we employed the IoU framework to evaluate the effectiveness of our "ONE" and "MINUS" models in detecting sound sources within spectrograms.

**Prediction on Test Images:**

Once our models are trained as described above, we use them to predict the "ONE" and "MINUS" labels on a set of test images containing spectrograms depicting diverse sound sources, including emissions from Wave Energy Converters (WECs). The images attached below are a sample of the predictions of the models.

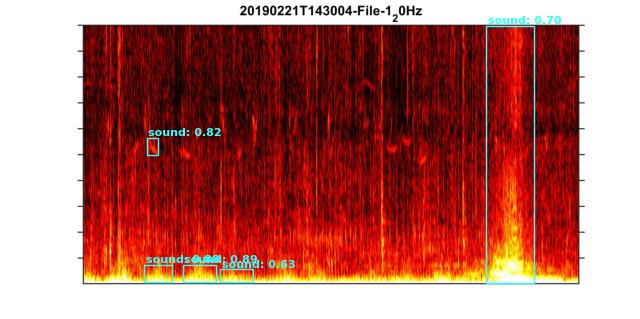


Figure 5: “One” Model Predictions

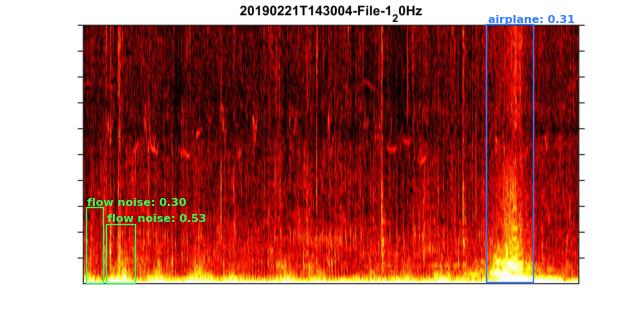
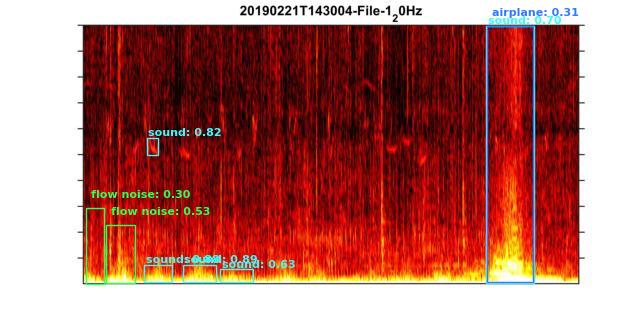


Figure 6: “Minus” Model Predictions

**Intersection over Union Calculation:**

Following model predictions, we conduct an intersection over union calculation by overlapping the "MINUS" labels on the "ONE" image. This process enables us to identify areas of agreement or disagreement between the models in identifying sound sources. The figure attached below shows the overlapping of the “One” and “Minus” predictions.

Figure 7: Overlapping of the “One” and the “Minus” Model Predictions

**Identification of Areas of Interest:**

Through the calculation of IoU scores for each bounding box pair, we identify areas where the agreement between the "ONE" and "MINUS" models is below a predefined threshold (e.g., 20%). These areas, marked as areas of interest, signify regions where the models diverge in their predictions or face challenges in identifying sound sources accurately.

**User Evaluation and Determination:**

Users are provided with bounding boxes delineating the areas of interest, allowing them to visually inspect and determine whether these regions contain noises emitted by WECs or other unidentified sources. The feedback and analysis provided by users play a crucial role in refining the models and enhancing their accuracy in identifying sound sources within spectrograms. The image attached below shows how the framework determines the area of interest.

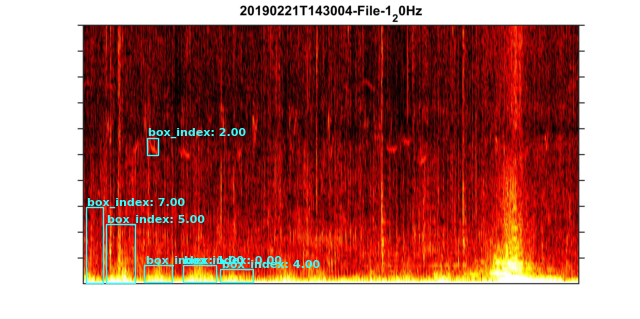


Figure 8: Areas of Interest the Marine Acoustic Experts can examine

In summary, the IoU framework serves as a valuable tool for assessing the performance of our "ONE" and "MINUS" models by quantifying the agreement between their predictions. By identifying areas of disagreement and marking them as areas of interest, we enable marine acoustics expert evaluation and refinement of the models for improved detection of sound sources, including those emitted by WECs.

**Recommendations**

In light of the outcomes from our “ONE MINUS” approach to classifying underwater acoustic signals, particularly the challenges presented by the "ONE" model's modest performance, we recognize the need for refinement in our methodologies. The future improvement of our model hinges on integrating both identified and unidentified noise sources within the same images during training. This will enable the model to better distinguish the specifics of identified sources and reduce the potential for confusion between them. Moreover, we propose the adoption of a single, unified model to process spectrograms. This change is anticipated to prevent the propagation of individual model errors into the final predictions, thereby reducing compounded error margins and enhancing overall identification accuracy.

**Resources**

[1] YOLOV8: a new State-of-the-Art Computer Vision model. (n.d.). https://yolov8.com/

[2] Ultralytics. (n.d.). GitHub - https://github.com/ultralytics/ultralytics