“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.

**Abstract**

This project explores the application of object detection capabilities to classify sounds within spectrograms, with a specific emphasis on identifying sounds emitted by Wave Energy Converters (WECs). Given the challenge of a lack of training data of the diverse sound profiles of various WEC devices, we introduce the "ONE MINUS" method. This innovative approach employs two distinct YOLOv8 models: the "ONE" model, which detects all discernible sounds under a unified "sound" label, and the "MINUS" model, which provides more detailed classifications.

**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 sounds emitted by Wave Energy Converters (WECs). Acknowledging the impracticality of compiling comprehensive training data for all possible representations of WEC sounds, due to the unique and variable sound profiles generated by differing WEC devices, our team devised the “ONE MINUS” method. This dual-model strategy employs two distinct YOLOv8 models: the “ONE” model, which detects all discernible sounds using a unified "sound" label, and the “MINUS” model, which classifies sounds with more granular labels. The aim was not to isolate WEC sounds directly but to filter out known sounds, thereby reducing the workload for marine acoustics specialists in identifying WEC emissions from the remaining unidentified sounds.

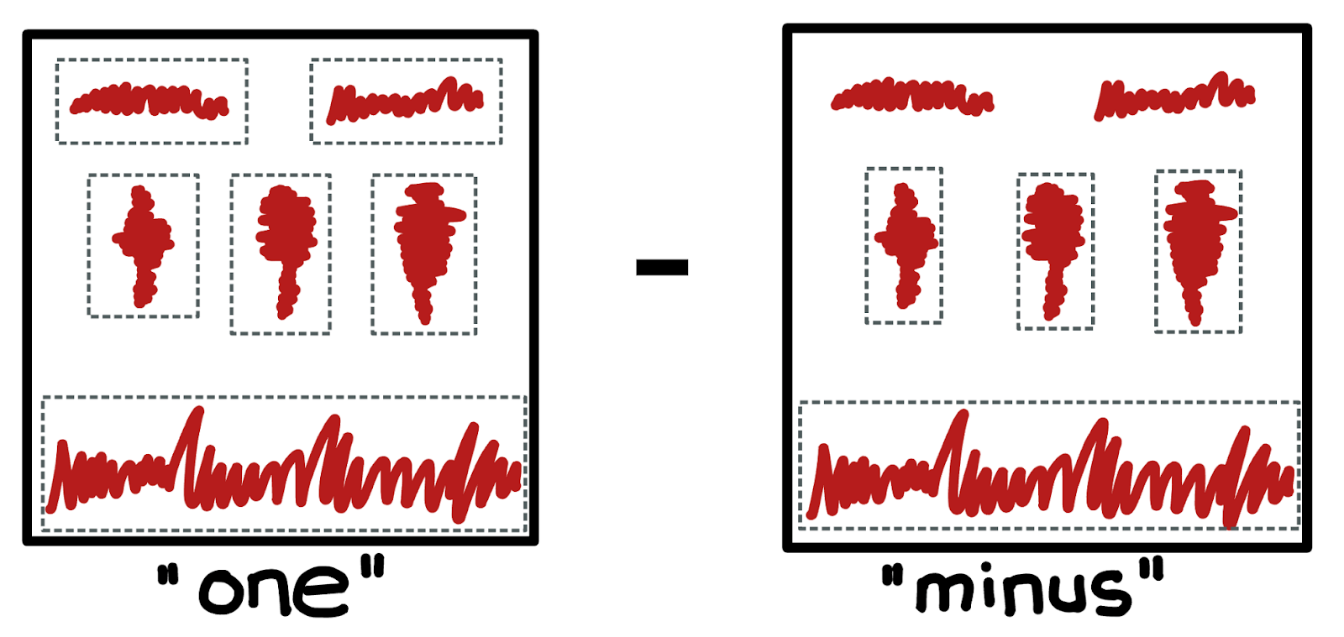


Figure 1: Visual representation of the “One Minus” approach

Object detection is particularly advantageous for applications requiring the identification of objects of interest within a scene, without necessitating precise location or shape delineation.

The primary objective of our project is to accurately identify sounds emitted by Wave Energy Converters (WECs) within a spectrogram. However, the task is challenged by numerous factors that hinder the generation of comprehensive training data for all possible spectrogram representations of WEC sounds. Variations in mechanical and architectural designs among WEC devices result in unique sound profiles for each device. Furthermore, these sound signatures are subject to change over time due to wear and the impact of underwater conditions on mechanical components.

To overcome these challenges, our approach incorporates a dual-model strategy utilizing the YOLOv8 framework, which we describe as the “ONE MINUS” method. This methodology involves the training of two distinct YOLOv8 models: the “ONE” model is designed to detect all discernible sounds within the spectrogram, whereas the “MINUS” model aims to classify as many known sounds as possible. Subsequently, for each spectrogram, we compare the output bounding boxes generated by the “ONE” model against those from the “MINUS” model. By excluding the known sounds identified by the “MINUS” model from the “ONE” model’s outputs, we isolate the unidentified sounds for further examination. These residual sounds, not recognized by the “MINUS” model, are then subjected to manual review to ascertain whether they can be attributed to WEC emissions.

This sophisticated approach enables us to navigate the complexities associated with the acoustic diversity and variability of WEC devices, enhancing our ability to detect and classify WEC-related sounds within spectrograms accurately.

**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

**How to read a Spectrogram?**

The provided spectrogram serves as a visual representation of audio signals. Identified by marine acoustics experts, the labeled elements within the spectrogram highlight distinct sound patterns and frequencies.

* The X-axis represents the time dimension (1 minute in the sample)
* The Y-axis represents the frequency bands
* The color of the waves represents the intensity of the audio signal. The brighter the waves, the louder the audio signal.

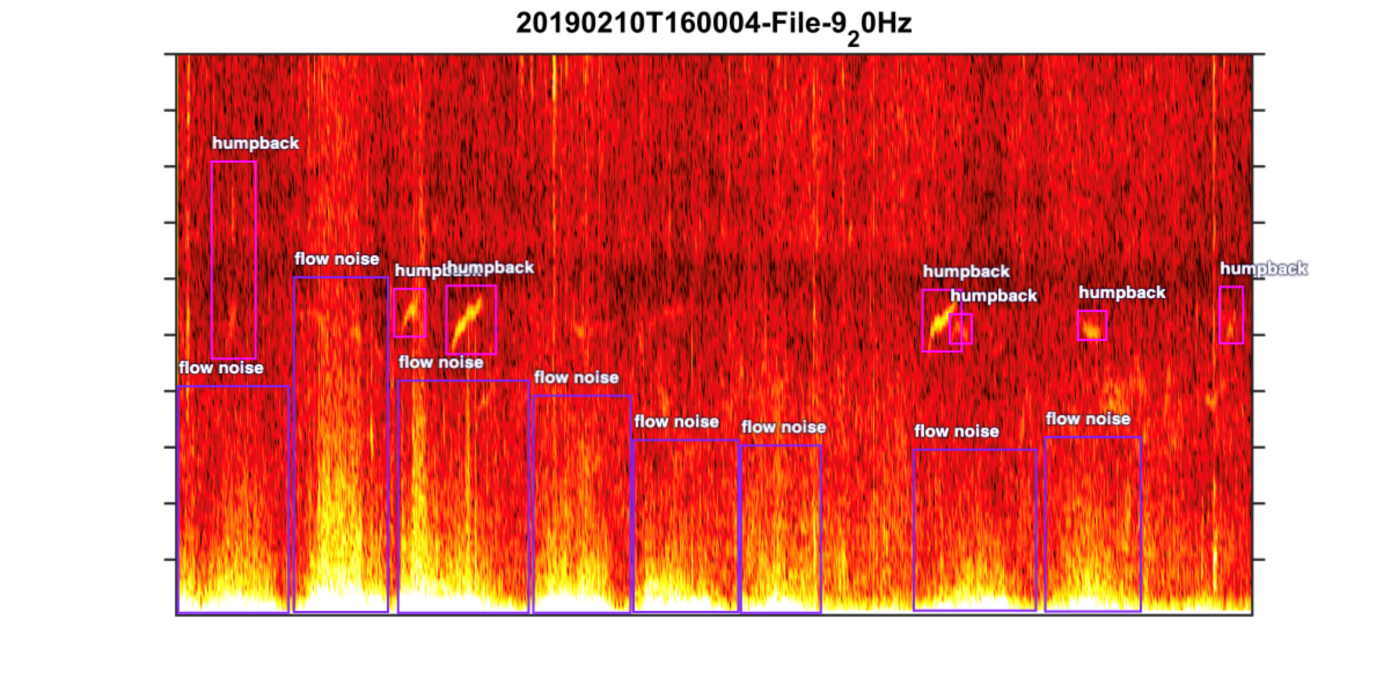


Figure 2: Sample Spectrogram

Platforms Used

We used several platforms through our project for various purposed.

**Roboflow for Annotation:**

* Roboflow provided a user-friendly interface for annotating data, allowing for the efficient labeling of spectrograms and associated sound classifications.
* Its annotation tools facilitated the creation of labeled datasets essential for training the object detection models.

**Azure ML for Hyperparameter Tuning Experiments:**

* Azure Machine Learning enabled the orchestration of hyperparameter tuning experiments in parallel.
* Leveraging Azure ML's capabilities, we efficiently explored various configurations to optimize the performance of the YOLOv8 models.
* The platform's scalability and automation features streamlined the experimentation process, accelerating model refinement.

**Google Colaboratory for Code Development and Testing:**

* 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.

By integrating these platforms into our project workflow, we enhanced productivity, scalability, and collaboration, enabling us to effectively tackle the challenges of sound classification within spectrograms.

**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.

In pursuit of a collaborative tool for annotating Dataset B, and one that could facilitate the import of Dataset A in LabelMe format, we selected Roboflow. For Dataset A, we utilized annotation classes including Humpback, Flow Noise, Airplane, Helicopter, Boat, and Mooring. For Dataset B, the classes were slightly adjusted to include Humpback, Flow Noise, Airplane, Helicopter, Boat, Interesting, and WEC.

**Annotation Classes and Conversion**

Using Roboflow, we converted annotations from Datasets A and B into 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. Through Roboflow's dataset versioning feature, we tailored these annotations for each model's training needs.

For the "MINUS" model, we excluded certain classes like Mooring due to the complexity of consistent annotation. This ensured that the model was trained on a streamlined set of classes relevant to our objectives.

For the "ONE" model, we simplified the classification by aggregating all detailed class labels into a single "sound" label. This approach was intended to aid the "ONE" model in detecting any sound within the spectrogram without distinguishing between specific classes.

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. This method facilitated efficient training by maintaining the full spectrum of annotations while adapting them to the specific needs of each model.

**Model Training and Configuration**

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. This decision was informed by a balance of performance and efficiency, evidenced by our testing which revealed a significant difference in runtimes between GPU (approx. 30 minutes for 100 epochs on Google Colab) and CPU (approx. 2.5 hours on Azure ML CPU clusters) processing.

**Model Selection and Evaluation**

In selecting the foundational model, we evaluated various YOLOv8 models, ultimately choosing yolov8m.pt for its balance of performance and efficiency. Performance testing 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) processing.

**Hyperparameter Tuning**

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

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **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: Commencing with the yolov8m.pt model, we first trained on Dataset A to obtain a refined model, subsequently integrating Dataset B for further enhancement.
* Integrated Training: Starting with the yolov8m.pt model, we concurrently trained on both Datasets A and B, aiming for a unified model optimization from the outset.

This methodical approach to model development and optimization underscores our commitment to advancing sound classification within spectrograms, specifically targeting the identification of WEC-related sounds. Through meticulous data preparation, collaborative annotation, and strategic model training, we aim to contribute valuable insights and methodologies to the field of acoustic signal processing.

**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. However, when the model was trained on a combination of Dataset A and Dataset B, there was a marginal decrease in performance, with a mAP50 score of 0.806. The Confusion Matrix for the results of the “Minus” model is attached below.

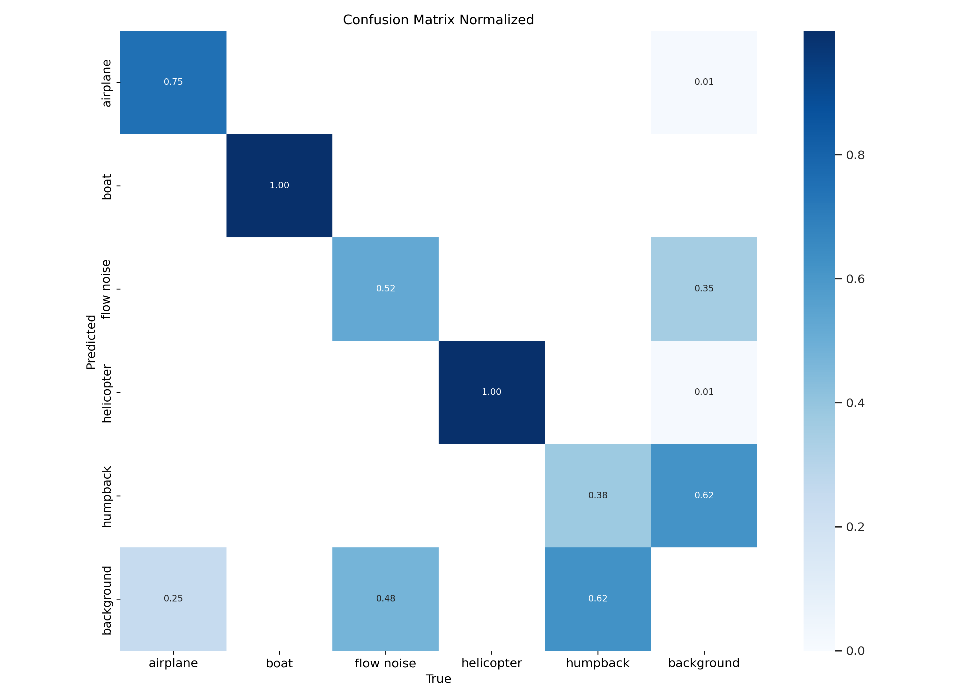


Figure 3: Best “Minus” Model Confusion Matrix

**“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. Combining the 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.

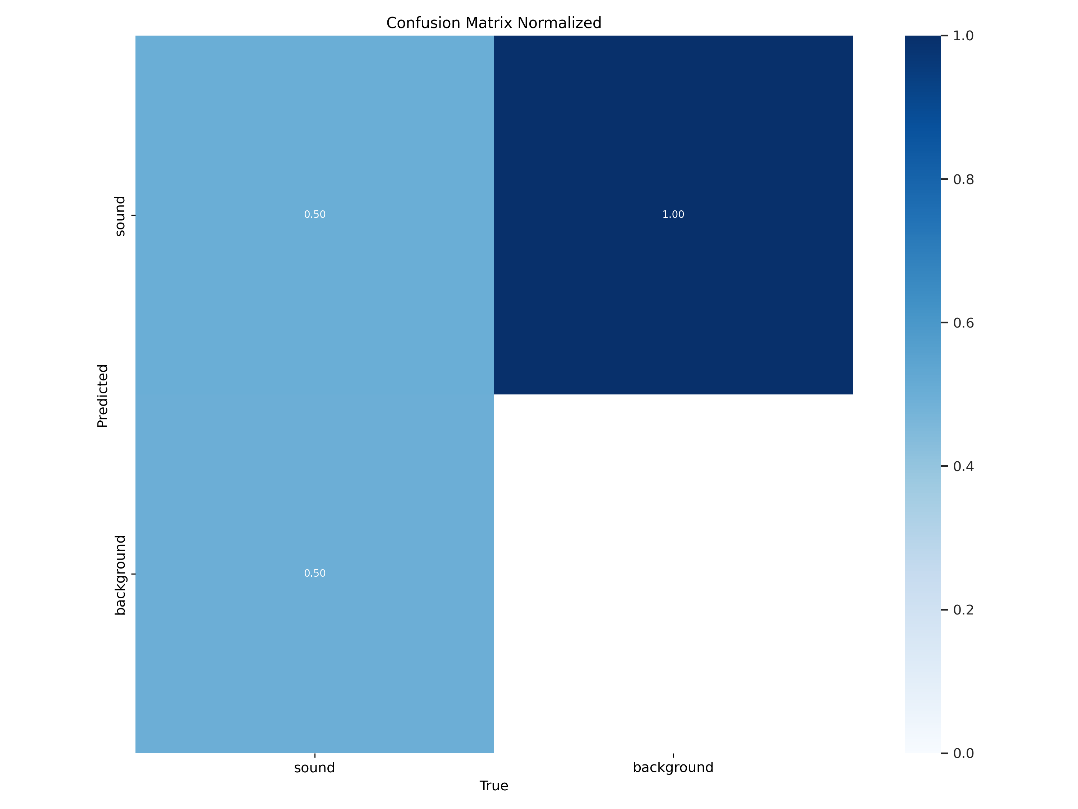


Figure 4: Best “One” Model Confusion Matrix

**Interpretation**

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 it could mean that Dataset B introduces complexity or variability that the models find challenging to generalize from 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 employ the IoU framework to evaluate the effectiveness of our "ONE" and "MINUS" models in detecting sound sources within spectrograms.

**Model Training and Prediction:**

Initially, we train our "ONE" and "MINUS" models using the best-trained weights obtained during the training phase. These models are specifically trained to identify various sound sources present within spectrograms.

**Prediction on Test Images:**

Once our models are trained, we utilize 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) and other unidentified sources. The images attached below are a sample of the predictions of the models.

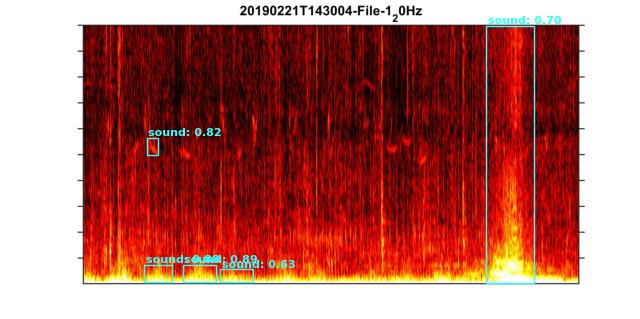


Figure 5: “One” Model Predictions

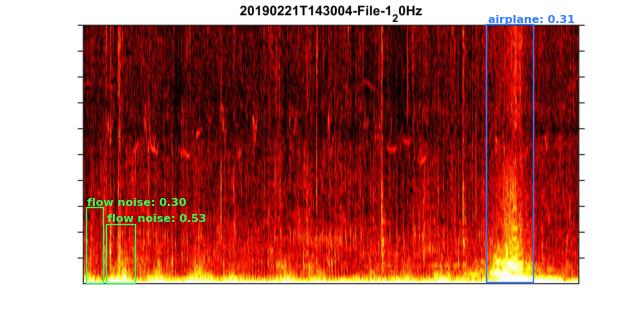
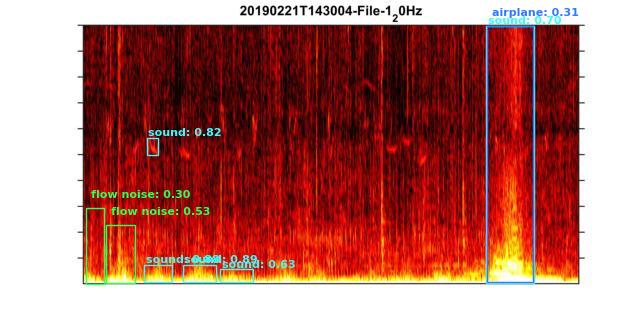


Figure 6: “Minus” Model Predictions

**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.

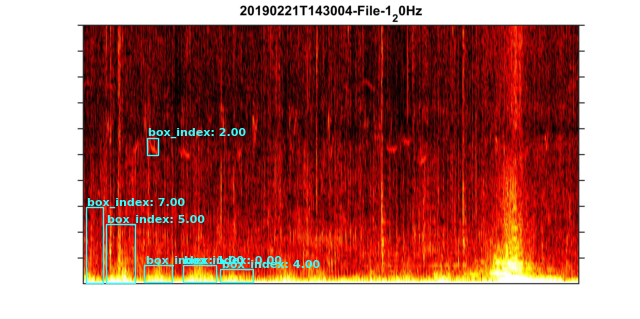


Figure 8: Areas of Interest the Marine Acoustic Experts can examine

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.

This framework not only provides the bounding boxes of signals that could potentially be WEC noises but it also finds and labels signals that could belong to unidentified sources. Using the coordinates of the boxes we provide; the marine acoustics experts could also map back to the time stamp of the audio signal from the spectrogram and examine the audio file.

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