Bioacoustics Detection on Marine Animals from Cal Poly Pier

Sophia Chung | spchung@calpoly.edu

Anagha Sikha | arsikha@calpoly.edu

Sucheen Sundaram | sssundar@calpoly.edu

Client: Professor Maddie Schroth-Glanz | mschroth@calpoly.edu

Advisor: Dr. Hunter Glanz | hglanz@calpoly.edu

Advisor: Dr. Jonathan Ventura | jventu09@calpoly.edu

I. Abstract

Our research aims to advance passive acoustic monitoring (PAM) along the Central Coast of California, particularly at the Cal Poly Pier in Avila, through the development of machine learning models for detecting and classifying marine animal signals. Building upon past work, we use an ensemble of Variational AutoEncoders (VAE) trained on Cal Poly Pier hydrophone recordings, as well as different preprocessing and postprocessing techniques to enhance detection capabilities. After all the changes we made to the pipeline, we achieved a low accuracy of 0.00214. To improve our process and performance metrics, we began implementing two new detection methods, Convolutional Neural Network and Detection Transformer. We hope our results contribute to advancements in signal detection, and that students who continue this project explore both detection and classification methods to further understand marine animals and their ecosystems, specifically the impact of human activities on underwater habitats.

II. Background

Motivation

PAM is essential for understanding and preserving marine ecosystems by allowing scientists to listen to the acoustics of marine animals and identify the sources of sound in underwater environments. This bioacoustics research contributes to the advancement of PAM along the Central Coast of California. Students on the Marine Acoustics Research Team at Cal Poly, led by Professor Maddie Schroth-Glanz, have spent several hours listening to underwater audio files to detect and to classify animal signals. However, manually detecting and classifying sounds is inefficient and laborious. This time-intensive process hinders the analysis of acoustic data in order to better understand migration patterns, feeding, reproduction, and other known behaviors. Thus, to streamline the current manual process, previous data science teams have developed a machine learning pipeline to detect marine animal signals based on manually annotated data. Our research aims to refine and enhance this pipeline for detecting marine animal signals collected specifically from the Cal Poly Pier in Avila by using larger amounts of data and fine-tuning parameters to produce desirable model metrics.

Data

The Marine Acoustics Research Team has collected data from hydrophones placed at two site locations: Monterey Bay and the Cal Poly Pier at Avila. The data collected are audio files ranging from two to three hours each from Monterey Bay and 30 minutes each from Cal Poly Pier, stored in Amazon Web Service (AWS). There are three deployments from which data was recorded from the Cal Poly Pier hydrophone. Deployment 1 consists of files that are in Glacier storage (currently inaccessible to save on costs), and Deployment 3 does not have decimated audio files, which is the type of file we use as input to our pipeline. Thus, we focused on using the 706 decimated audio files from Deployment 2. However, only 39 of these files were accompanied by corresponding annotation files, providing a smaller subset for comparison and analysis. These annotation files are text files that delineate the time and frequency ranges within the audio files where marine animal signals are detected, along with the species responsible for each signal.

Classification is another important aspect of this research; however, Deployment 2 contained data with only one label, gray whale. Meanwhile, the data in Deployment 1 is in the process of being manually classified by species.

2023 Team's Work

The most recent data science team made great efforts to improve detection through an unconventional method of training a model to denoise audio recordings from the Monterey hydrophone, and thus isolating the animal signals. They utilized a sole background noise file, which was created by removing animal signals from a hydrophone recording, as the basis for their training data. By leveraging manually created labels indicating the presence of signals in these recordings, the team was effectively able to strip away noise. Unlike traditional object detection approaches, their model concentrated on denoising audio, training the model on what not to include.

The team used preprocessing techniques in their model: Short-Time Fourier Transform (STFT) and Per-Channel Energy Normalization (PCEN). STFT analyzes frequency over time and attempts to localize frequency. PCEN enhances the visibility of animal signals by balancing volume across different energy bands and minimizing model bias to specific frequencies. As seen in the spectrograms of Figure 1, these techniques combined visually reduced the presence of background noise and distinguished animal signals.

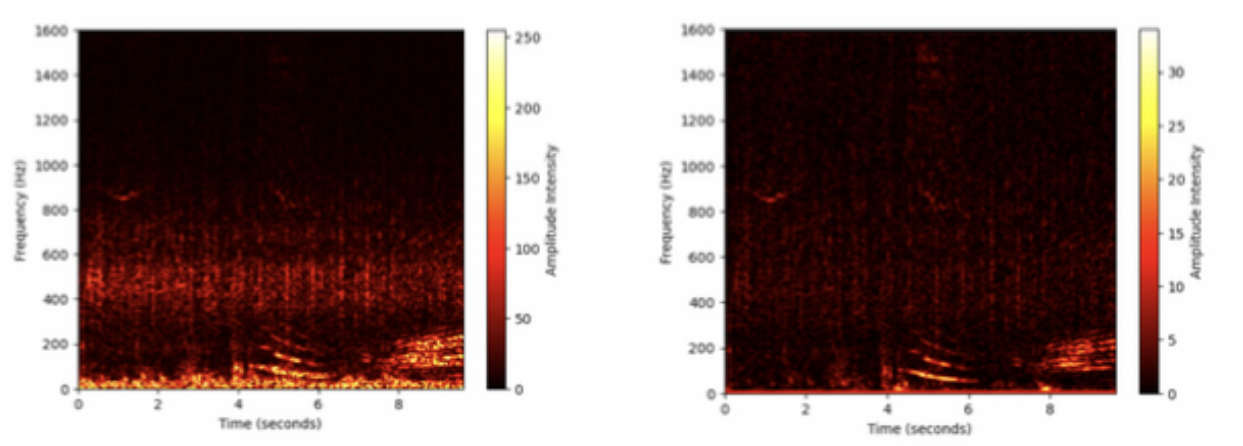

Figure 1. Pre-PCEN vs. Post-PCEN in reducing presence of background noise and visually enhancing signals

The team employed a VAE model which followed a multi-step process, as seen in Figure 2. The decimated background noise was provided as the training input, teaching the model to recognize normal acoustic patterns. Subsequently, the probabilistic encoder transformed the audio data into a compressed representation known as latent space. The latent space served as a simplified version of the original audio data, allowing for efficient processing. Next, the probabilistic decoder reconstructed the audio data from the latent space, aiming to closely resemble the original input while removing background noise. For this project, the output serves as the denoised version of the input audio, where the background noise has been effectively reduced while preserving the essential characteristics of the original sound.

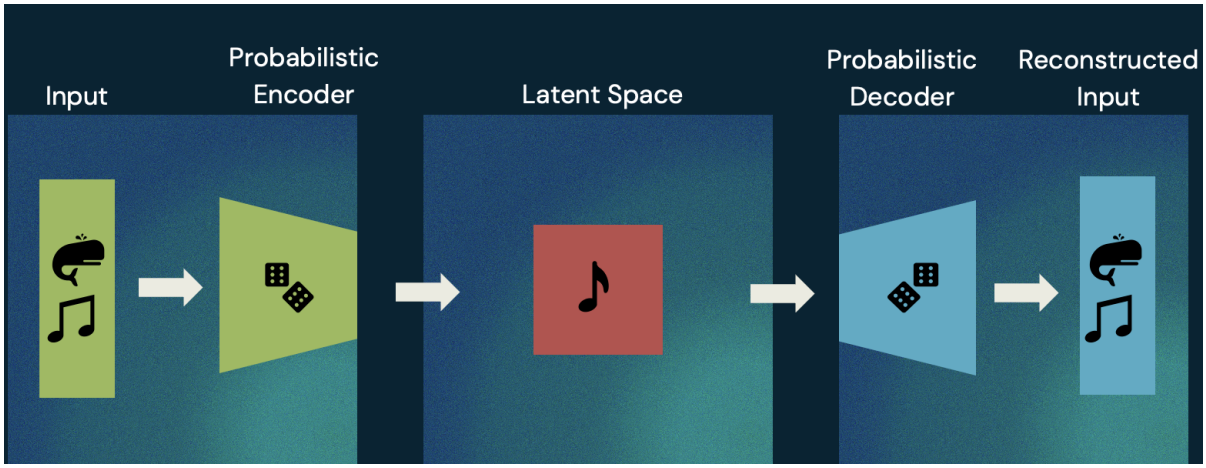

Figure 2. Process of Variational AutoEncoder

Object detection pipelines like this one typically have postprocessing methods to refine their predictions. Non-maximum suppression (NMS) is a common technique to do so. The team implemented their interpretation of NMS, which involved analyzing pairs of predicted boxes that have an overlap exceeding a given threshold. For each pair, only the box with the higher confidence is kept. Because they used an ambiguous method for calculating confidence that we deemed illogical and because NMS depends on confidence, we decided not to use this technique and to create our own, which we detail in the next section.

III. Methodology

2023 Team's Code

We ran the 2023 team's code on the Cal Poly Pier hydrophone recordings in order to train the VAE model on new data. This allowed us to get the necessary output files and practice running on existing architecture. To follow this process, we first needed to create a background noise file for the Cal Poly Pier data. We randomly selected a hydrophone recording from the Cal Poly Pier data and used an existing labels file to strip any animal signals from the recording. After getting the final training background noise file, we ran their modeling code to train the initial VAE on a single file. The output of this model was a set of bounding boxes on a file of our choosing. The prediction's file consisted of the timestamps of the detected signal and the respective frequencies.

Preprocessing: Mel

To refine our preprocessing methods, we researched common preprocessing techniques for audio deep learning models. We found that it is common to convert frequency from the standard hertz scale to the mel scale when dealing with audio data within human hearing range. The mel scale was developed to account for the fact that humans perceive frequencies on a logarithmic scale rather than a linear scale like the hertz scale. We confirmed with our client that we were only expected to detect signals within the human hearing range, since there are several marine animal signals that cannot be naturally detected by humans. This conversion makes signals much more apparent, as is visualized in Figure 3.

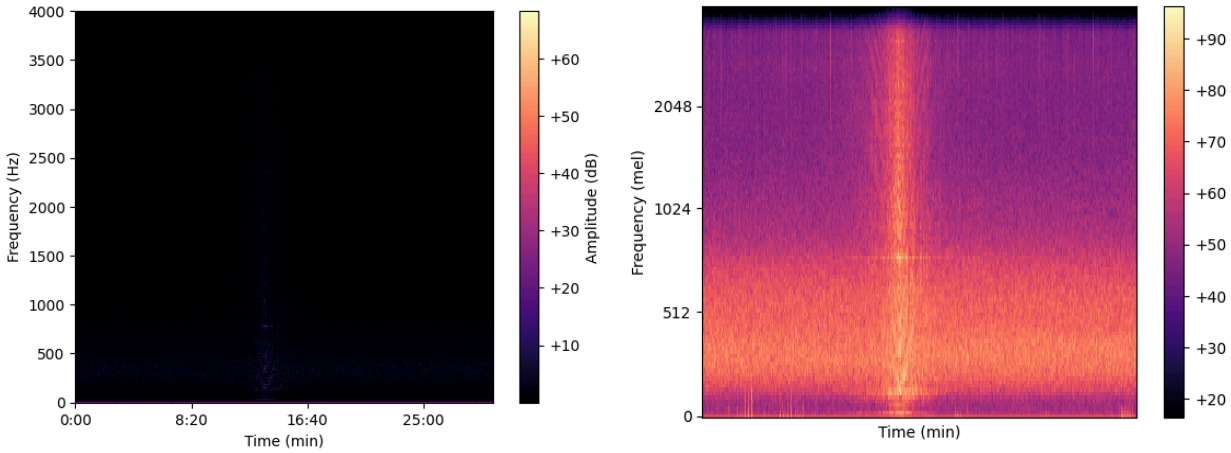

Figure 3. Hertz vs. Mel scales in revealing signals

Model: VAE Ensemble

The 2023 team did not use much data to train their model. We wanted to train on larger amounts of data, so we created an ensemble of VAEs utilizing all Cal Poly Pier hydrophone recordings. We wanted to make a more generalizable model that we could use on many different hydrophone recordings. To implement this, we trained on all background noise files from the Cal Poly Pier data. We also applied the team's technique of STFT and PCEN as preprocessing techniques, which is the final form of the training data. We then improved the VAE model by dividing the training work due to computing constraints from the data size. An ensemble model was the best solution to combat this issue because we could train each unit in the ensemble on a smaller chunk of the data. This technique ensures that the size of the training data for each model in the ensemble is manageable. The ensemble consisted of 10 variational autoencoders. Each unit was trained on approximately two hours of data, or 4 files each, totaling to 39 files. The ensemble method outputs predicted bounding boxes of signal timestamps and their respective frequencies. When predicting an unseen hydrophone recording, we got each model's output bounding boxes and stored that in a data frame.

Postprocessing: Box Combination

To refine our predictions, we believe it would be best to use a clustering algorithm because we want to remove duplicate boxes as well as combine boxes that are very similar. We chose to use agglomerative hierarchical clustering over other methods because it allows flexibility with the amount of boxes per cluster and the similarity level we want. After clustering, we were left with clusters of similar boxes. To combine all of the boxes in a cluster into one final box, we calculated their intersection. The intersection represents the area that was picked up by multiple predictions. The alternative to this was taking the union of all the boxes, but this would result in larger areas of larger uncertainty, which is not ideal.

Model Training: Amazon SageMaker

Due to computing constraints, we used Amazon SageMaker, a cloud-based machine-learning platform, to train our VAE ensemble. We needed a powerful kernel to process extensive background noise files and store all the intermediate steps of the model, which led us to use the

largest available kernel (ml.m5d.24xlarge) with 384 GB of RAM, 96 virtual CPUs, and 900 GB of instance storage. Using this instance, we trained the ensemble model on all 39 files in 100 minutes, compared to the six hours it would have taken using our local machines. Although training the VAE ensemble and producing predictions on SageMaker cost $450, a majority of our $500 budget, we obtained our results quickly. Using cloud computing highlighted the balance between obtaining fast results and managing expenses effectively.

IV. Results

Intersection over union (IoU) is a measure that is used to evaluate the performance object detection pipelines. It is calculated as the intersected area divided by the total area of the two boxes. We use this measure to classify our predictions as follows:

- A true positive is a predicted box that has an IoU larger than a certain threshold with an annotated box.
    - If two predicted boxes have a large enough IoU with the same annotated box, two true positives are counted.
- A false positive is a predicted box that does not have a large enough IoU with an annotated box.

A false negative is an annotated box that does not have a large enough IoU with a predicted box, and in this context, we do not consider true negatives as our focus is on identifying signals and not their absence.

We used these classifications to produce metrics for each of the 39 files, both before and after using our box combination technique. We then averaged them for easy comparison in Table 1.

Table 1. Comparing Average Metrics

	Number of Predicted Boxes	Accuracy	Precision	Recall	F1
Before Box Combination	503.64103	0.00076	0.00077	0.00675	0.00137
After Box Combination	177.64103	0.00214	0.00216	0.00651	0.00302

V. Discussion

Prior to implementing our box combination technique, there were 20 times the amount of predictions than annotations, on average. There is a noticeably large improvement in the metrics from before we implemented our box combination technique to after: the average number of predicted boxes was reduced by about 65%. Our new accuracy was 0.00214, which means that out of all predicted boxes, 0.214% matched an annotated box. Our new recall was 0.00651, which means that our pipeline correctly identifies 0.651% of annotated boxes. We realize that we should have used a weighted average of our metrics since some files contain many more signals than others, so that is something to keep in mind regarding interpretability of our results.

Despite successful implementation of the mel scale in preprocessing, which was verified visually, we were not able to retrain our VAE ensemble with new inputs due to computational limitations and our budget. As long as it is appropriate though, we do recommend maintaining this step of the preprocessing, as it is widely used and conceptually sound.

Although the 2023 team switched to a VAE model with the primary goal of reducing background noise, they ran into the same issue of getting too many predictions. From our metrics, we believe that even with our updates, a VAE framework is not suitable for this data and task. Thus, we pivoted to exploring new models for detection: a Convolutional Neural Network (CNN) and a Detection Transformer (DETR).

Alternative Model: 2022 Team's CNN Model

CNNs are well-suited for object detection and classification tasks because they use convolution instead of matrix multiplication in their layers. This makes CNNs effective for image-related tasks as they can learn spatial hierarchies and accurately capture both local and global patterns, which is crucial for interpreting complex data like audio signals. Upon revisiting work from a team in 2022, we found that they successfully implemented a CNN in their black box model. Their approach, in contrast to the VAE ensemble, took the raw audio files as input without stripping any animal signals. This leverages the CNNs strength in handling unprocessed data and extracting meaningful features.

The 2022 team split up the signal detection work into three notebooks. The first notebook focuses on producing spectrograms representing the collection of decimated audio files. The second notebook focuses on building, tuning, and training the models using SageMaker's object detection algorithm. The third notebook focuses on using the spectrograms and trained models to assess model performance and predict the locations of animal signals in audio files. We initiated the process of running the 2022 team's CNN model on the Cal Poly Pier data; however, due to time constraints, we have not yet obtained any output. Our shift to using CNNs is based on their proven effectiveness in similar image-related tasks, offering a promising alternative to improve detection accuracy.

Alternative Model: Hugging Face Object Detection Guide

Hugging Face is an open-source collaborative platform for artificial intelligence. One of the computer vision guides they offer is for object detection. The guide uses a model called DETR, which combines a convolutional backbone with an encoder-decoder transformer, aspects of the 2023 team's VAE model and the 2022 team's CNN model.

With the limited time we had remaining upon deciding to switch frameworks, we attempted to set up an 80-10-10 training-validation-test split and properly format our data for input to the pipeline. Each input image is represented as a dictionary, with one of the fields being "bbox," which stands for bounding box. This field must follow the COCO format: [x, y, width, height] where x and y are the coordinates of the top left corner of a bounding box and all values are measured in pixels. COCO stands for Common Objects in Context, which is a large-scale object detection dataset that is widely referenced in the field.

Hugging Face provides the guide as a Google Colab notebook. We attempted to run their example locally, but with basic Google Colab resources, the notebook was projected to run for 40 hours. At this point in our project, we had already reached our budget, but we would recommend for other students to continue this work using AWS computing resources.

VI. Conclusions

In conclusion, we successfully implemented a VAE ensemble model using the Cal Poly Pier data to detect animal signals. However, our initial VAE model struggled with excessive predictions, as seen by the low accuracy, recall, precision, and F1 metrics. Implementing a box combination technique improved our results, but the performance remained suboptimal, suggesting that using a VAE ensemble was not suitable for this data and our tasks.

This led us to pivot from a VAE ensemble to more advanced models for object detection, CNN and DETR, in order to continue enhancing the detection accuracy of animal signals. Revisiting the 2022 team's work with CNNs showed us a promising alternative due to the CNNs' ability to capture spatial hierarchies and handle raw audio data effectively. Additionally, exploring the DETR model from the Hugging Face object detection guide offered another effective approach by combining a convolutional backbone with an encoder-decoder transformer. Although time and budget constraints prevented us from fully implementing these new models, we believe these methods demonstrate the potential for significant improvements in detection for the animal signals. Overall, we hope our findings contribute to the advancement of PAM efforts, offering valuable insights into the distribution and density of marine animal activity in the Avila ecosystem.

VII. Future Work

As this is an ongoing project, we have a few suggestions for future work that can build on our initial findings. Future teams should prioritize completing the implementation of the two detection methods, CNN and DETR, using cloud computing resources to handle the computational demands efficiently. Continuing to work on the two detection methods and obtaining results from these models will provide valuable insights into their effectiveness on detection accuracy. Additionally, there is potential for further optimization of these models based on their performance metrics.

Another important aspect of this project is the classification of the detected sounds. While our current efforts focused on detection, the natural progression is to classify the identified signals to determine the species or sources of the sounds. Unfortunately, we did not have access to data with species labels, which prevented us from incorporating classification into our project. We hope that future teams will have access to labeled datasets and can implement classification methods to enhance the project, thus enabling more detailed and informative analysis of the acoustics.

By continuing these efforts, future teams can build a comprehensive system that not only detects but also accurately classifies marine animal signals, contributing valuable data to the fields of marine biology and conservation.

VIII. Acknowledgements

We would like to thank Professor Schroth-Glanz for her guidance throughout this project. We would also like to thank the student researchers who worked on annotating the data. Lastly, we would like to thank Dr. Ventura and Dr. Glanz for their support throughout this project.

IX. References

"DETR." *Hugging Face*, huggingface.co/docs/transformers/model_doc/detr.

Doshi, Ketan. "Audio Deep Learning Made Simple (Part 2): Why Mel Spectrograms Perform Better." *Medium,* Towards Data Science, 18 Feb. 2021, towardsdatascience.com/audio-deep-learning-made-simple-part-2-why-mel-spectrograms-perform-better-aad889a93505.

Doshi, Ketan. "Audio Deep Learning Made Simple: Sound Classification, Step-by-Step." *Medium*, Towards Data Science, 18 Mar. 2021, towardsdatascience.com/audio-deep-learning-made-simple-sound-classification-step-by-step-cebc936bbe5.

"Evaluate." COCO, Common Objects in Context, cocodataset.org/#detection-eval.

Gammal, Nick, et al. "Detecting Marine Acoustic Profiles With Deep-Learning Denoising."

K, Sambasivarao. "Non-Maximum Suppression (NMS)." *Medium*, Towards Data Science, 1 Oct. 2019, towardsdatascience.com/non-maximum-suppression-nms-93ce178e177c.

"Object Detection." *Hugging Face*, huggingface.co/docs/transformers/en/tasks/object_detection.