WHALE-VAD: WHALE VOCALISATION ACTIVITY DETECTION

Technical Report

Christiaan M. Geldenhuys
cmgeldenhuys@sun.ac.za

Günther Tonitz
25040863@sun.ac.za

Thomas R. Niesler
trn@sun.ac.za

Department of Electrical and Electronic Engineering,
University of Stellenbosch, South Africa

ABSTRACT

In this work, we present a sound event detection (SED) system focused on whale call detection. We propose a hybrid CNN-BiLSTM architecture adapted from the voice activity detection (VAD) field in order to perform coherent per-frame whale call activity detection. In addition, we investigate the multi-objective regression task of bounding box estimation in conjunction to activity detection. We compare the performance of our system to a baseline mel spectrogram BiLSTM, and finetuned HuBERT system. As part of the 2025 BioDCASE challenge (Task 2), we also compare our system to ResNet-18 and YOLOVv11 models. Each model has been trained on a subset of the publically available ATBFL dataset. Our model was able to best all models including the top performing YOLOv11 model on developmental results. The final model, trained using the collapsed labels along with phase information, achieved an F1-score of 0.44, across all developmental sets.

***Index Terms*—** Whale Call Detection, Computational Bioacoustics, Sound Event Detection, Hybrid CNN-BiLSTM

1. INTRODUCTION

Passive acoustic monitoring (PAM) has enabled researchers to monitor species in remote locations using non-invasive and relatively low-cost methods. However, the large volumes of data generated can be difficult to process, especially due to the typically low signal-to-noise ratio (SNR) of such datasets [1]. This makes manual review both time-consuming and expensive, requiring the need for trained experts to annotate the large quantities of data.

To address this, many automated algorithms have been developed to detect and classify signals of interest. One focus has been the detection and classification of blue and fin whale vocalisations. Blue whales were driven nearly to extinction by the 20th-century whaling, and as a result are still considered endangered today. Together with fin whale, they are considered vulnerable by the IUCN red list [2], [3]. Population densities for both species remain difficult to estimate with confidence due to limited data availability [4].

Therefore, in this paper, we focus on developing a lightweight whale call activity detector based on a convolutional bidirectional long short-term memory neural network (CNN-BiLSTM) based on a voice activity detection (VAD) framework originally proposed for speech [5]. We explore a range of input features and demonstrate that incorporating phase information from the short time Fourier transform (STFT) can enhance model performance.

2. EXPERIMENTAL STRUCTURE

In this section, we describe the experimental setup used for sound event detection (SED) on the challenge dataset. Figure 1 illustrates the complete whale call activity detection system, used for sound event detection (SED). First the audio is segmented and preprocessed, each segment is provided to the model as input, with frame-level classification targets produced from the boundary annotation file. Each model consists of two parts: a *feature extractor*, that produces high-dimensional vectors representative of the information contained in the audio segment; and a *classification model*, tasked with producing a class membership probability from feature vectors, obtained at each time instant.

2.1. Preprocessing

The training data consists of long continuous audio recordings, typically the result of PAM. Due to computational constraints, these longer audio recordings were subdivided into shorter intervals, referred to as segments.

Each segment corresponds to the audio between the start and end points of a human annotation, indicating the occurrence of a particular call type. The segment is extended to include additional audio before the start and at the end of the call, referred to as a collar. The length of the collar is independently and randomly sampled from a uniform distribution for both the start and end of each segment, ensuring the call does not always appear in the centre of the segment.

An associated discrete classification target vector is constructed from the annotations for each segment. In all experiments, one model classification is computed every $20\,\mathrm{ms}$. As there may be overlapping annotations, the problem is treated as multi-class multi-label. Consequently, a binary label is assigned for each respective class at each discrete time instant, independent of the other classes. If a human annotation boundary intersects completely with the classification target vector at a time instant, the label is set to *true* indicating the presence of a particular call; otherwise, it remains *false*.

Additionally, segments without any vocalisation annotations are included (Section 2.5). In such cases, the entire classification target is set to *false*, indicating that no vocalisation has occurred.

The variable-length segments are gathered into a batch of fixed length during preprocessing. To achieve this, each segment and the associated classification target is padded to the length of the longest segment in the batch. However, padding is removed from each segment during loss calculation and model weight backpropagation.

During evaluation (either validation or testing), when human annotations are not available, segments are generated by subdividing the continuous audio recordings into regularly spaced segments

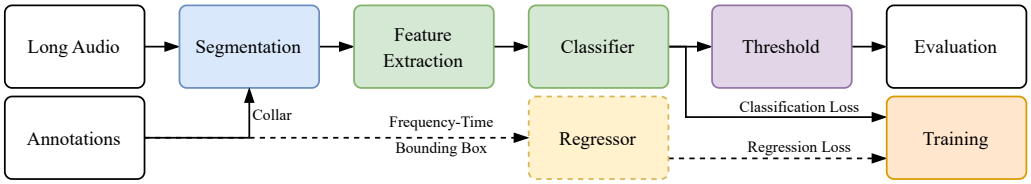

Figure 1: Illustration depicting the whale call activity detection system overview and experimental setup.

with a fixed length of 30 s and a 2 s overlap. The postulated model classification probabilities are averaged over this overlap.

2.2. Feature extractor

The feature extraction model is provided with the entire segment of audio samples obtained from a part of the larger continuous audio recording (Section 2.1). In this work, we only consider spectral and cepstral features.

First, a spectrogram representation is computed using a 1 s (approx.) frame length and 20 ms stride between frames. The long frame length was motivated by the low fundamental frequency of the whale calls. The frame stride was dictated by the desired classification resolution, which was fixed at 20 ms, to allow for direct comparison of loss figures between experiments (Section 2.1). A Hanning window is applied to each frame, without additional zero padding. A 256-point fast Fourier transform (FFT) is computed, resulting in 129 frequency bins (incl. DC) – resulting in the final spectrogram representation.

The power spectrum is compressed into 64 bins using a bank of triangular filters with a mel-scale spacing. The final sequence of mel frequency cepstral coefficients (MFCCs) are obtained by applying the discrete cosine transform (DCT) to the resulting binned spectrum, and retaining the lower 20 coefficients.

After the features are obtained, mean spectral and cepstral subtraction is performed, respectively. The mean is computed for each frequency bin independently, over the entire duration of the segment.

2.3. Baseline classification model

From the sequence of features obtained from the feature extractor, the call posterior probabilities are computed using a classification model with sigmoid activation functions at the outputs. We evaluate two classification model architectures: logistic regression (LR) and bidirectional long short-term memory networks (BiLSTMs).

In early experiments, LR was utilised primarily as a means of optimising dataset and experimental hyperparameters. Informally, we observed that LR models struggled to maintain high call probabilities over the entire duration of a call segment ($\theta > 0.5$). Specifically, these models would perform well for start-point detection, but they would not perform well in call endpointing.

As a recurrent architecture, a BiLSTM model was chosen and configured with between one and four hidden layers; a hidden dimension size of 64, 128, or 256; and layer dropout of between 20 % and 50 %. We found that the recurrent model was prone to overfit, but that increased dropout reduced this risk. The posterior call probabilities produced by these models aligned well with the call segments. Informally, it was observed that both the start and end boundaries produced by the recurrent models closely matched those of the human annotators.

2.4. Whale vocalisation activity detector

We propose a whale call activity detection system inspired by the AVA-VAD system proposed in Wilkinson and Niesler [5]. We alter the AVA-VAD system by introducing a residual bottleneck network and a depthwise convolutional neural network (CNN). Furthermore, instead of using a mel spectrogram as input, we utilise the spectrogram features directly and apply a linear convolution layer to act as a learnable filterbank. Figure 2 provides an overview of the model architecture.

The spectrogram is computed using the same configuration as, described in Section 2.2. However, during experimentation, we found that including phase information substantially improves detection performance. Thus, instead of the power spectrum typically utilised as feature, we provide the model with a three-dimensional representation of each complex spectral component (z) as follows:

$$z = r(\cos\theta + i\sin\theta); \qquad \boldsymbol{x}_k^{(n)} = \begin{bmatrix} r \\ \cos\theta \\ \sin\theta \end{bmatrix}$$

where r is the spectral magnitude and θ is the phase. Now $\boldsymbol{x}_k^{(n)}$ is the model input at time instant n and discrete frequency k.

The learnable filterbank consists of a linear convolutional layer. The one-dimensional kernel is convolved with each vector of energies constituting the spectrogram. This output is then passed through a two layer CNN with max pooling, GELU activation and batch normalisation. This is followed by a bottleneck network consisting of three convolutional layers, each with GELU activation, compressing the features to a lower dimensional representation. This representation is passed through three depthwise convolutional layers, with restricted cross-terms in the filter kernel. This configuration acts as a feature aggregation network. The output of the feature extractor, bottleneck network and depthwise convolutions are residually connected. Table 1 provides a summary of the CNN layer configuration employed by the final Whale-VAD model.

The residual output connection is passed through a linear dimensionality reduction layer with an output dimension of 64. Padding has been applied to each of the CNN layers such that the number of output activations remains consistent with the number of input frames. These latent features are then recurrently processed by a BiLSTM network. Finally, a linear layer with sigmoid activations produces the model call probabilities.

We further investigate two input regularisation techniques: spectral augmentation [6] and noise perturbation. For noise perturbation, we inject Gaussian noise into the audio signal such that the resulting SNR of the original signal to the perturbed signal is 10 dB.

2.5. Stochastic negative mini-batch undersampling

Analysis of the challenge dataset revealed that whale vocalisations are a rare occurrence, with a prevalence of approximately 5 %. There-

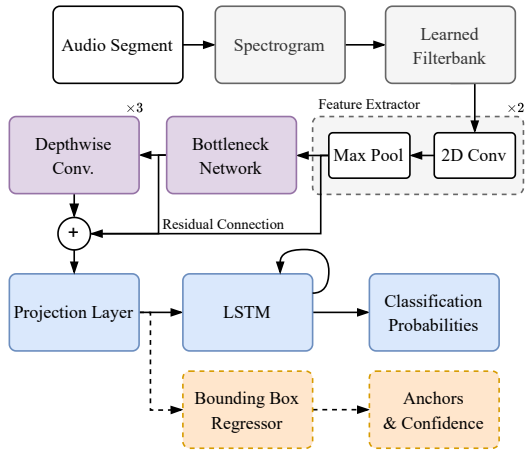

Figure 2: Illustration of Whale-VAD system, for whale call activity detection.

Table 1: Summary of Whale-VAD model layer configuration. The kernel size (K), stride (S), number of input channels (C_{in}) and output channels (C_{out}), are shown.

Layer	K	S	C_{in}	C_{out}
Filterbank	(7, 1)	(3, 1)	1	64
Feature extractor				
└ Conv2D	(5, 5)	(3, 1)	64	128
└ Max pool	(5, 1)	(1, 1)	–	–
└ Conv2D	(3, 3)	(2, 1)	128	128
└ Max pool	(3, 1)	(1, 1)	–	–
Bottleneck network				
└ Conv2d	(1, 1)	(1, 1)	128	64
└ Conv2d	(3, 3)	(1, 1)	64	64
└ Conv2d	(1, 1)	(1, 1)	64	128
Depth. Conv2d	(3, 3)	(1, 1)	128	128

fore, we propose a technique where, during each epoch of finetuning, we sample a different subset of *negative* segments (containing no calls) while simultaneously ensuring that there are approximately as many negative as positive segments, per mini-batch. This ratio was anecdotally found through a limited set of experiments comparing the binary cross entropy loss (non-weighted) on the validation set for different subsampling values, using a LR classifier. Furthermore, after each epoch of training, a different subset of negative segments is sampled. The set of positive calls remains consistent for each epoch during finetuning.

2.6. Loss function

For our experiments, we consider weighted binary cross-entropy (BCE) and *focal loss* as the chosen loss functions. We found that when computing the class weighting, rather than normalising by the duration of each class, it was better to normalise by the number of segments belonging to each class. When considering weighted BCE, we compute the weighting w_c for each class c as follows:

$$w_c = \frac{N}{P_c}$$

where N denotes the total number of negative (no-call) segments and P_c denotes the number of positive (call) segments belonging to a particular class c.

In addition to weighted BCE, we also evaluate the use of *focal loss* [7], a modified cross-entropy loss designed to focus training on *hard-to-classify* examples by reducing the contribution of easy examples. In our experiments, we set the class imbalance term to 0.25 and *focus* term to 2, following the recommendations in the original paper.

For all experiments, we rely on AdamW [8] as the numerical optimiser. Unless otherwise stated, the optimiser was configured with an initial learning rate of 1×10^{-5}, momentum terms of 0.9 and 0.999, and a weight decay factor of 0.001.

2.7. Transfer learning

Beyond our baseline (MFCC + BiLSTM) and Whale-VAD models, we investigate the efficacy of a finetuned HuBERT [9] model (initialised with pretrained weights) and a classifier (Section 2.3) to achieve whale call activity detection. The HuBERT model is an end-to-end automatic speech recognition (ASR) model that contains a CNN feature extractor and transformer encoder with self-attention. The HuBERT has roughly $95\,\mathrm{M}$ parameter weights, while our baseline has $125\,\mathrm{k}$ and Whale-VAD $1.1\,\mathrm{M}$. HuBERT natively requires a sampling rate of $16\,\mathrm{kHz}$ and thus the audio was upsampled to match. Finetuning was performed using transfer learning with a learning rate scheduler in order to attempt to preserve knowledge built up during the pretraining phase and to reduce the chance of overfitting [10]. In this transfer learning process, the classification model (initialised with random weights) is allowed weight updates for the first five epochs of training, while the HuBERT backbone is not allowed any weight updates. During this period of model training, the backbone acts as an embedding model. All early stopping logic is suspended until this stage. After which, the backbone model weights are updated using a learning rate $10\,\%$ of the current learning rate of the classification model. Over the next two epochs, the learning rate is increased gradually to match the learning rate (global) of the classification model.

2.8. Multi-objective regression

The challenge dataset contained not only annotations in time, but also in frequency (bounding box). The best-performing baseline YOLO model, provided by the organisers, uses these box-level annotations. In addition to our Whale-VAD system (Section 2.4), we evaluated a bounding box regression network. The network is based on the same latent features used by the classification model, with the addition of an adaptive pooling layer in order to reduce the time dimension. These reduced latent features are presented to two independent multi-layer perceptron (MLP) networks, each consisting of three layers, with GELU activation and dropout after each hidden layer. Each of the regression networks is applied to the 64 channels from the adaptive pooling layer, which is the maximum number of anchors (bounding boxes) the model can produce per input segment. The first network has a four dimensional output, corresponding to the bottom left and top right corners of the bounding box. The second network produces a confidence score, corresponding to the presence of the bounding box. Figure 2 illustrates the additional bounding box regression model. The regression model is trained using smoothed L1 loss function [11]. Note that the regression model only forms part of a multi-objective training regime, where the regression and classification loss are jointly optimised. The regression outputs (bounding

Table 2: Final development set results for the organisers models ResNet18 and YOLOv11, baseline MFCC, HuBERT, AVA-VAD and Whale-VAD models. Scores are averaged across all call types and validation sets. *Improvement* is calculated relative change in F1 score.

Experiment	Improvement	Recall	Precision	F1-score
ResNet18	–	0.36	0.29	0.32
YOLOv11	–	0.32	0.67	0.43
MFCCs + BiLSTM	–	0.409	0.226	0.291
HuBERT + BiLSTM	–	0.064	0.104	0.079
AVA-VAD [5]	–	0.310	0.219	0.245
Whale-VAD	–	0.424	0.207	0.278
+ Phase information	+30.0%	0.461	0.316	0.375
└ + Noise perturbation	+1.2%	0.413	0.335	0.370
└ + Augmentation	-13.5%	0.391	0.335	0.361
└ + Bounding box reg.	-22.1%	0.380	0.262	0.310
└ + Focal loss	+8.0%	0.484	0.348	0.405
└ + Three class problem	+15.2%	0.461	0.420	**0.440**

boxes) are not used for final model evaluation. We postulate that training the network to jointly optimise the model on both tasks may lead to improved classification performance.

2.9. Postprocessing

After model training is completed, the best model is chosen based on the lowest BCE validation loss. The classification thresholds θ_c are selected per class c, from the precision-recall curve on a held out set, and the point chosen with the best F1-score on the held out set. The resulting per call threshold θ_c is applied to the posterior call probabilities computed by the model, to obtain the final binary model output labels. The output labels are then post-processed to reduce overlap, fragmentation, and duplication.

The 7-class output of the classifier is collapsed into the 3-class variant, posed by the challenge organisers. While this approach may occasionally merge overlapping calls of different subcall types, such occurrences are infrequent in the training and validation datasets and are considered an acceptable trade-off for improved overall accuracy.

The resulting binary labels are then used to generate annotations with start and end boundaries relative to the start of the recording. These annotations are further refined by merging overlapping calls of the same type, eliminating duplicates, and joining calls separated by less than $500\,\mathrm{ms}$ to reduce instances where an event is momentarily missed by the model. Finally, calls that are either too long or too short are discarded, based on duration constraints derived from empirical statistics.

3. RESULTS

It is evident from Table 2 that both spectral and cepstral models significantly outperform the finetuned HuBERT model. Furthermore, a series of enhancements to the spectral CNN-BiLSTM architecture led to notable performance gains. Most significantly, incorporating phase information resulted in a $30\,\%$ improvement in F1-score. Training the model on collapsed labels yielded an additional $15.2\,\%$ increase, achieving a final F1-score of 0.440. Table 3 and Fig. 3 provides a detailed break-down of our best performing model's results per development set.

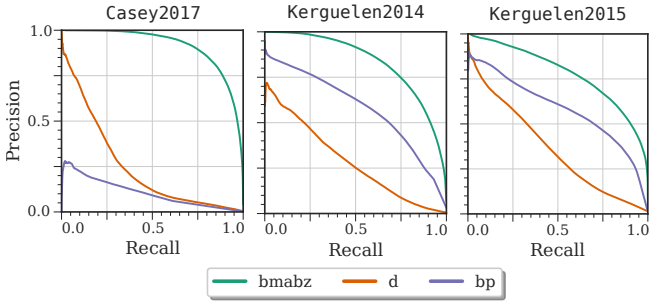

Figure 3: Precision-recall curves for the validation sets using the top-performing Whale-VAD model.

Table 3: Detailed development set results for the top-performing Whale-VAD model using the trigonometric complex representation as input features and training with the collapsed labels using weighted binary cross-entropy (BCE).

Dataset	Label	TP	FP	FN	Recall	Precision	F1
casey2017	bmabz	1984	1956	434	0.821	0.504	0.624
casey2017	d	179	5928	374	0.324	0.029	0.054
casey2017	bp	5	101	287	0.017	0.047	0.025
kerguelen2014	bmabz	2739	1120	1558	0.637	0.710	0.672
kerguelen2014	d	229	2248	550	0.294	0.092	0.141
kerguelen2014	bp	1391	663	2355	0.371	0.677	0.480
kerguelen2015	bmabz	2137	2676	611	0.778	0.444	0.565
kerguelen2015	d	366	2545	1158	0.240	0.126	0.165
kerguelen2015	bp	665	355	605	0.524	0.652	0.581

4. CONCLUSION

During development, several finetuned self-supervised transformer models were explored, including HuBERT. While these models showed some promise, none achieved performance comparable to that of the simpler detector-based models, despite their very large number of parameters and success in other tasks. We attribute this gap primarily to the low sampling rate of the recordings, which is only $250\,\mathrm{Hz}$. Given that the pretrained transformer models were developed for audio sampled at $16\,\mathrm{kHz}$, it is likely that the $250\,\mathrm{Hz}$ sampling rate lacks sufficient frequency content for accurate call detection, with the base architecture.

While the majority of prior research on SED has disregarded phase information in the STFT, considering it to be redundant in speech processing tasks, we have demonstrated that incorporating it for whale call classification can improve model performance by as much as $30\,\%$ compared to models trained solely on magnitude features, alone. This finding opens avenues for future research in bioacoustics to reconsider the role of phase in time-frequency representations, particularly in the design of feature extractors and model architectures that can more effectively exploit both magnitude and phase components.

5. ACKNOWLEDGMENTS

The authors gratefully acknowledge the computing time provided to them on the Stellenbosch Rhasatsha high performance computing (HPC1) facility. We thank the contributors of the open-source software package *PyTorch* [12] for developing this key tool.

Christiaan M. Geldenhuys orcid.org/0000-0003-0691-0235
Güther Tonitz orcid.org/0009-0009-6030-4122
Thomas R. Niesler orcid.org/0000-0002-7341-1017

6. REFERENCES

[1] K. A. Kowarski and H. Moors-Murphy, "A review of big data analysis methods for baleen whale passive acoustic monitoring," *Marine Mammal Science*, vol. 37, no. 2, pp. 652–673, 2021, ISSN: 0824-0469, 1748-7692. DOI: 10.1111/mms.12758.

[2] J. G. Cooke, "Balaenoptera musculus," *The IUCN Red List of Threatened Species*, 2018, Erratum published in 2019.

[3] J. G. Cooke, "Balaenoptera physalus," *The IUCN Red List of Threatened Species*, 2018.

[4] B. S. Miller, The IWC-SORP/SOOS Acoustic Trends Working Group, K. M. Stafford, *et al.*, "An open access dataset for developing automated detectors of antarctic baleen whale sounds and performance evaluation of two commonly used detectors," *Scientific Reports*, vol. 11, no. 1, p. 806, 2021, ISSN: 2045-2322. DOI: 10.1038/s41598-020-78995-8.

[5] N. Wilkinson and T. Niesler, "A hybrid CNN-BiLSTM voice activity detector," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, Canada: IEEE, 2021, pp. 6803–6807, ISBN: 978-1-7281-7605-5. DOI: 10.1109/ICASSP39728.2021.9415081.

[6] D. S. Park, W. Chan, Y. Zhang, *et al.*, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Graz, Austria, 2019, pp. 2613–2617. DOI: 10.21437/Interspeech.2019-2680.

[7] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, *Focal loss for dense object detection*, 2018. DOI: 10.48550/arXiv.1708.02002.

[8] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proceedings of International Conference on Learning Representations (ICLR)*, Toulon, France, 2017.

[9] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, *HuBERT: Self-supervised speech representation learning by masked prediction of hidden units*, 2021. DOI: 10.48550/arXiv.2106.07447.

[10] C. Vasconcelos, V. Birodkar, and V. Dumoulin, *Proper reuse of image classification features improves object detection*, 2022. DOI: 10.48550/arXiv.2204.00484.

[11] R. Girshick, *Fast r-CNN*, 2015. DOI: 10.48550/arXiv.1504.08083.

[12] A. Paszke, S. Gross, F. Massa, *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS)*, New Orleans, USA, 2019, pp. 8026–8037.