NAVE: A NORMALIZED, ADAPTIVE CONFORMER FOR ANTARCTIC WHALE VOCALIZATION-EVENT DETECTION

Technical Report

Matthias Nagl

Student at
Johannes Kepler University Linz, Austria
k1126326@students.jku.at

ABSTRACT

We present NAVE, a single Conformer detector built around a front-end that combines per-bin complex mean subtraction, which removes the stationary spectral background of long underwater recordings, with a per-channel energy normalisation (PCEN) channel that adapts its gain to the local noise floor. These feed a frequency-dynamic stem and a Conformer whose depthwise convolution is sized to the duration of the target calls. We apply NAVE to BioDCASE 2026 Task 2, the strongly-labelled detection of Antarctic blue and fin whale calls across three call types, BMABZ, D, and BP. Under leave-one-site-out evaluation the single model exceeds the published staged pipelines for this task, predicting events from a single forward pass and a per-class threshold, with no boundary proposals and no post-processing search. A controlled ablation shows that the front-end, not the detector head, recovers the D downsweep, the faint, background-buried call that most systems on this task detects least well. It also shows that the wide, duration-matched depthwise kernel yields the largest further gain.

***Index Terms*—** Sound event detection, bioacoustics, PCEN, Conformer, receptive field

1. INTRODUCTION

Passive acoustic monitoring (PAM) is an important, non-invasive tool for studying Antarctic blue and fin whales, which roam vast, remote stretches of ocean and are difficult to observe directly [1, 2]. It produces large volumes of low signal-to-noise audio whose manual annotation is slow and expert-dependent [3]. Automatic detection is therefore attractive but hard: calls are sparse, conditions vary strongly between sites, and the same call type looks different with distance, noise, and instrumentation. These properties make BioDCASE Task 2 a useful benchmark for whale-call sound event detection [4, 5].

The task is strongly labelled multi-label detection: a system returns both the class and the temporal bounds of each event. The seven original call labels are grouped by acoustic similarity into three evaluation classes [6]: BMABZ for blue-whale A-, B-, and Z-calls, D for blue- and fin-whale downsweeps, and BP for fin-whale 20 Hz pulse calls. The development set is drawn from the Acoustic Trends Blue Fin Library (ATBFL) of annotated Antarctic recordings [7, 5].

Two dataset properties shaped our design. First, the data are highly imbalanced in time: only a small fraction of the audio carries call annotations, so false positives quickly dominate the event counts and reducing them matters almost as much as detecting the calls. Second, the merged class distributions vary strongly across site-years, and D in particular is concentrated in a few training sites, drawn largely from elephantisland2013 and elephantisland2014. D is also the rarest and most recall-limited class, and has remained the bottleneck for prior detectors on this data [8, 9].

Recent systems improve on the WhaleVAD CNN-BiLSTM detector [8] along two axes that leave the backbone unchanged. One acts on the output stage: WhaleVAD-BPN adds a boundary proposal network that gates the classifier output, with a systematic optimisation of the post-processing [9]. A second acts on the training data, through hard-negative mining and semi-supervised consistency training on unlabelled audio [10, 11]. We ask whether the backbone is a comparable lever, and present NAVE: a Conformer [12] fed by a normalised, adaptive front-end, with a depthwise convolution whose receptive field is matched to call duration. A single such model is competitive with the stronger published baseline. A controlled ablation then points to the input features as the most likely cause of D's remaining difficulty: varying the architecture, kernel width, loss, and input context length all left D essentially unchanged. Improving D will therefore take richer features or more labelled data, not a larger backbone. Our contributions are:

- We combine a frequency-dynamic stem [13] and a fixed PCEN channel [14] with a Conformer backbone and show the combination is competitive as a single model.
- We show the Conformer's depthwise kernel acts as a duration-matched receptive field, and characterise where widening it stops helping [15].
- We document a battery of controlled negatives across architecture, kernel, loss, and input context, none of which improved D, which we read as evidence that its bottleneck lies in feature sensitivity.

2. METHOD

Figure 1 gives an overview of the NAVE detector, which addresses the detection task defined by the challenge (Section 2.1). What is new in NAVE is the front-end and the Conformer backbone. Its segmentation and event post-processing follow WhaleVAD [8] unchanged, so that our ablations isolate the architecture from the surrounding pipeline.

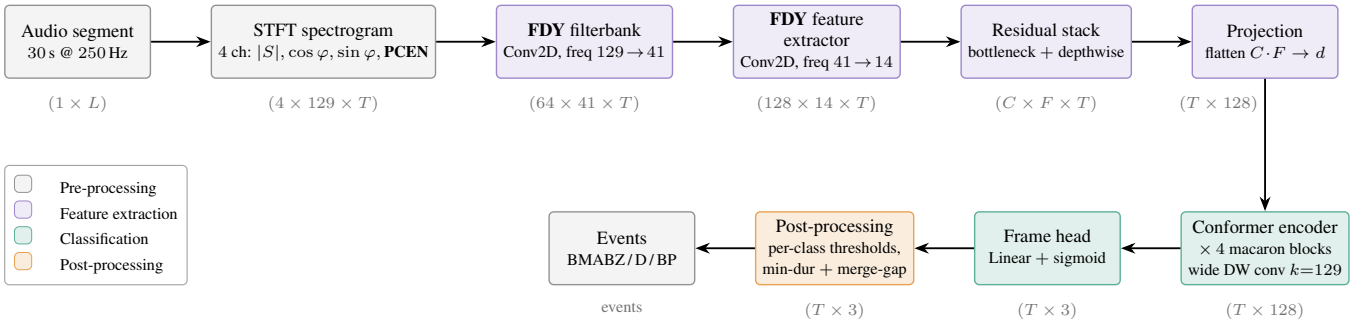

Figure 1: Overview of the NAVE detector. A four-channel phase-aware spectrogram with a fixed PCEN channel feeds an FDY convolutional stem and a four-block Conformer whose convolution module uses a wide depthwise kernel (k=129). A per-class sigmoid head produces frame posteriors that are post-processed into events.

2.1. Task and SED formulation

We treat the detection task as frame-level multi-label classification. Each recording is divided into fixed-length segments (Section 3), and within a segment the model emits, for each of the three classes, an independent posterior sequence at a resolution of one decision per 20 ms. Because calls of different types may overlap in time, the three class posteriors are produced independently through sigmoid activations rather than a softmax over classes.

The per-frame targets are built from the boundary annotations: a frame is labelled positive for a class if it falls within an annotated call of that class, and negative otherwise. Segments with no annotated call are retained as negatives, since calls are sparse and the model must learn to stay silent over the long background stretches that dominate the recordings.

At inference the recording is tiled into regularly spaced, overlapping segments and the per-frame posteriors are averaged over the overlap into a single stream per recording, which the post-processing of Section 2.6 converts into events scored as in Section 3.

2.2. Normalised, adaptive front-end

Each segment is transformed by a short-time Fourier transform with a 256-sample (approximately 1 s) Hann window and a 5-sample (20 ms) hop, giving 129 linear frequency bins at a frame rate of one per 20 ms. The long window suits the low fundamental frequencies of baleen-whale calls. From the complex spectrogram we form a three-channel, phase-aware input. Each frequency bin of the complex spectrogram is mean-subtracted over the segment, which suppresses the stationary background while preserving the transient calls. The three channels are then the magnitude and the cosine and sine of the phase of this mean-subtracted spectrogram. This trigonometric phase representation follows WhaleVAD [8], which found it to improve detection on these calls.

To these three channels we append a fourth channel obtained by per-channel energy normalisation (PCEN) of the raw magnitude [14, 16]. PCEN replaces the usual logarithmic compression with an automatic gain control: each frequency bin is divided by a smoothed estimate of its own recent energy, formed as an exponential moving average with smoothing coefficient 0.025, before a compressive nonlinearity with gain 0.98, bias 2.0, and exponent 0.5. Because the gain adapts to the local background level, PCEN drives the slowly varying noise floor towards unity and lets transients that outrun the moving average stand out. It is aimed in particular at the weak, background-buried D-call downsweeps, the most difficult and recall-limited class. It raises the contrast on which the backbone then operates, without on its own separating the true downsweeps from the confusing transients it also lets through, which is where D stays limited (Section 4.1).

We use a fixed PCEN channel rather than a learnable one. A learnable PCEN and other learnable front-ends were evaluated and none improved on the fixed channel (Section 4.1). The resulting four-channel representation is passed to the convolutional stem described next.

2.3. Frequency-dynamic stem

NAVE's convolutional stem maps the four-channel spectrogram to a per-frame feature sequence through a filterbank convolution, which maps the frequency bins to a smaller set of learned bands, followed by a feature-extraction block, a bottleneck, and a depthwise aggregation stage. The stem follows the WhaleVAD feature extractor [8], except that we make its two earliest convolutions, the filterbank and the first convolution of the feature-extraction block, frequency-dynamic (FDY) [13]. We apply FDY only to these two, where the frequency axis is still finely resolved, and leave the rest of the stem standard. A standard two-dimensional convolution applies one shared kernel across all frequency bins, which presumes translation invariance along frequency. For these calls that presumption does not hold, because absolute frequency is class-informative: the three call groups occupy characteristic frequency ranges, so the same local time-frequency pattern means something different at 20 Hz than at 100 Hz. An FDY convolution instead maintains K=4 basis kernels and, for each output frequency bin, mixes them with weights from a softmax attention over the time-averaged input, so the effective kernel adapts to the band it operates on.

2.4. Conformer backbone and its receptive field

NAVE's temporal backbone is a Conformer encoder [12] of four blocks, each combining a macaron pair of feed-forward modules (one at each end of the block), multi-head self-attention with rotary positional embeddings [17], and a convolution module, at a model width of 128 with four attention heads. A per-frame linear layer reads out the three class logits at the 20 ms frame rate. The complete model has 2,693,499 parameters.

Within each block, the convolution module is a depthwise convolution whose kernel width sets how much contiguous temporal context a single layer integrates. At the 20 ms frame rate, the kernel of 129 frames spans 2.58 s. Self-attention already mixes information globally across the segment, so this wide kernel is not about temporal reach. Its role is to provide a fixed, local, translation-equivariant operator over a window long enough to span a call's time-frequency structure, the drawn-out BMABZ tonals and the D downsweep, a complementary inductive bias that the global, content-based mixing of attention does not supply on its own. Because the convolution is depthwise, widening the kernel adds only $d_{\text{model}} \times k$ weights, about 16,500 at k=129, so the receptive field can be enlarged at almost no parameter cost. We treat this kernel width as the single architectural lever of the backbone and sweep it in Section 4.1. The kernel is kept odd so that padding preserves the frame count.

2.5. Training

We optimise with RAdam [18], which is designed to remove the need for a learning-rate warmup. The complete training configuration is given in Table 1. The loss is a masked weighted binary cross-entropy following WhaleVAD [8]: padding frames do not contribute, and the positive term for each class c is scaled by $w_c = N/N_c$, where N_c is the number of positive segments for class c and N the total across the three classes, so the rarer classes contribute more. Because calls are sparse, each epoch pairs all positive segments with an equal number of randomly sampled no-call segments, redrawn at the start of every epoch.

We maintain an exponential moving average of the model weights and batch-normalisation statistics [19], and validate and checkpoint the averaged weights rather than the raw ones. Its role is to stabilise training against the per-epoch resampling of the negative pool, which can shift the operating point from epoch to epoch, most visibly for the sparse D class. Because we rely on this averaging, we keep the learning rate constant rather than decaying it, as weight averaging benefits from a non-decaying learning rate [19]. At each epoch the per-class detection thresholds are re-tuned (Section 2.6) and the checkpoint is selected by the resulting F1.

2.6. Post-processing

The per-recording posterior stream (Section 2.1), formed from 60 s tiles with 2 s overlap, is converted into events by a fixed pipeline following WhaleVAD [8]. These tiles are longer than the 30 s training segments because the added context improves detection, most clearly for BMABZ (Section 4.1). The stream is smoothed with a 500 ms median filter and thresholded per class. Adjacent same-class events are merged when separated by less than 0.5 s, and only events with a duration between 0.5 and 30 s are kept. The per-class thresholds are the only tuned component, fitted by coordinate descent. The smoothing window, merge gap, and duration bounds are fixed and shared across all classes, which limits the validation-set tuning to a single threshold per class.

3. EXPERIMENTAL SETUP

We use the BioDCASE Task 2 development set, drawn from the ATBFL introduced in Section 1. Eight site-years form the training set (ballenyisland2015, casey2014, elephantisland2013, elephantisland2014, greenwich2015, kerguelen2005, maudrise2014, and rosssea2014) and three, casey2017, kerguelen2014, and kerguelen2015, form the validation set on which we select checkpoints and tune thresholds. The training configuration is given in Table 1.

Table 1: Training configuration. The architecture is described in Sections 2.2–2.4 and Figure 1.

Hyperparameter	Value
Optimiser	RAdam (weight decay 10^{-3}, grad clip 1.0)
Learning rate	5×10^{-5}, constant (no warmup)
Batch size	32
Epochs	40
Loss	masked weighted BCE ($w_c = N/N_c$)
Segment length	30 s
Negatives	ratio 1.0, resampled each epoch
Weight averaging	EMA (decay 0.999)
Dropout	0.1
Checkpoint selection	tuned F1

Every configuration is trained with three seeds. The ablations of Section 4.1 use a single seed to isolate each design choice, and we report the mean over the three seeds.

Scores are computed at the event level. Detections are matched to the reference annotations by one-dimensional temporal intersection-over-union at a threshold of 0.3, and the true positives, false positives, and false negatives are pooled over the three validation sites to give a precision P_c and recall R_c for each of the $C = 3$ classes. We report F1 as the harmonic mean of the class-averaged precision and recall, the metric computed by the official challenge evaluation,

$$\mathrm{F1} = \frac{2\bar{P}\bar{R}}{\bar{P}+\bar{R}}, \qquad \bar{P} = \frac{1}{C}\sum_{c=1}^{C} P_c, \qquad \bar{R} = \frac{1}{C}\sum_{c=1}^{C} R_c. \quad (1)$$

4. RESULTS

4.1. Development

All development numbers are on the three validation sites at the 60 s operating point (Section 2.6), with the per-class thresholds tuned on the same split. They are optimistic in absolute terms but consistent for ranking, which is their purpose here. They are single seed, and repeated runs vary by about 0.03 in F1 (the three-seed range on the generalisation split, Section 4.2), so we read smaller differences as indicative and draw conclusions from the larger steps and from per-class behaviour rather than from the aggregate alone.

Table 2 adds one component at a time on a fixed seed. The decisive step is the PCEN channel, which lifts D from 0.181 to 0.299 with BMABZ unchanged, a front-end sensitivity change rather than a backbone or loss change, and the cleanest causal result in the development. The normalised channel is what recovers the weak, background-buried D downsweeps. This jump is single seed and a repeated run reached a smaller value, so we report it as a strong direction that further seeds should confirm. The wide depthwise kernel is the largest single step in aggregate, raising both BMABZ and D. The frequency-dynamic stem, the first rung, gives a gain that sits inside the seed band on its own, but it is the substrate the later components build on.

The wide kernel rewards further widening. With the training held fixed, sweeping the depthwise kernel from k=65 to 161 at a

Table 2: Development ladder on the three validation sites (single seed, 60 s tiles, official F1 of Eq. 1). Rows 1–4 each add one component, the base being the Conformer of Section 2.4 (constant-rate RAdam and EMA) and the fourth a wide depthwise kernel. The final row is the complete model re-tuned at the chosen k=129, a separately optimised run, so its gain over the row above combines the wider kernel and the re-tuned training, not a single component.

System	BMABZ	D	BP	F1
Conformer base	0.391	0.106	0.418	0.329
+ FDY stem	0.362	0.181	0.478	0.351
+ PCEN channel	0.360	0.299	0.497	0.391
+ wide kernel (k=65)	0.523	0.377	0.533	0.479
NAVE (k=129)	0.569	0.456	0.521	**0.524**

fixed seed raises F1 monotonically, by about 0.03 over the range, a gain carried by BMABZ whose F1 climbs steadily with width while D and BP move within the seed band. Within that band the per-class optima still track call duration, with BP, the briefest call, strongest at the narrowest kernel, D peaking at an intermediate width, and BMABZ, the longest, gaining to the widest kernel tested. We read this as suggestive support for the duration-matched receptive field of Section 2.4. We fix the kernel at k=129 as a balance, near the top of the aggregate trend while avoiding the loss on D that appears at the widest kernel. At k=129 NAVE reaches an F1 of 0.524 at this seed (Table 2) and 0.542 at the best of several seeds.

A wide range of further changes did not improve on the final model. Frequency attention, a learnable LEAF front-end [20], a learnable per-band PCEN, a dual-resolution input, delta channels, multi-scale depthwise kernels under both summed and concatenated fusion, a temporal decoder head, and recall-tilted losses (asymmetric loss [21] and Focal-Tversky [22]) all regressed or tied within seed variation. Several of these targeted D specifically, and none moved it appreciably: the kernel sweep shifts D only within the seed band and non-monotonically, the tile-length sweep leaves D flat from 20 to 120 s while BMABZ rises with context, and the front-end sensitivity change of the PCEN channel is the one intervention that moved D at all. We therefore read D as bounded by feature sensitivity rather than by the backbone, kernel, loss, or context length: on this data only better features or more data improve it.

4.2. Generalisation and comparison

The development thresholds above are tuned on the same sites they are scored on. To estimate generalisation we re-select the per-class thresholds under leave-one-site-out cross-validation across the three sites. Thresholds are tuned on two sites and scored on the held-out third, and the three held-out scores are averaged. The trained model is identical across folds, only the thresholds are re-selected, and the reported result is the mean over the trained seeds, each scored independently under this protocol.

Table 3 sets NAVE against the published baselines under cross-validation. Averaged over seeds, the single NAVE model exceeds both, including the stronger WhaleVAD-BPN, while using neither its boundary-proposal stage nor its post-processing search and producing events from a single forward pass and a tuned per-class threshold. The three seeds are tightly grouped, with a standard deviation of 0.014 in pooled F1, and all exceed WhaleVAD-BPN. The lead is carried by D and BP, where the single model improves on both pipelines, while it absorbs a smaller shortfall on BMABZ, the class the staged pipelines are built to detect well. A single principled front-end and backbone is therefore competitive with, and here ahead of, the staged pipelines. That best seed and a twelve-seed, probability-averaged ensemble are our two submitted outputs.

Table 3: Cross-validated comparison on the three evaluation site-years, with per-class F1 and the official F1 of Eq. 1. WhaleVAD and WhaleVAD-BPN are the cross-validated results of [9]. For NAVE we report the single model averaged over seeds, pooled over the three sites and then per held-out site, with thresholds tuned on the other two. Bold marks the best of the three systems. Per-site F1 is computed on one site alone, so the pooled value is not the mean of the three.

System	BMABZ	D	BP	F1
WhaleVAD [8]	**0.628**	0.126	0.315	0.377
WhaleVAD-BPN [9]	0.615	0.340	0.460	0.475
NAVE, single model (mean)	0.557	**0.425**	**0.517**	**0.512**
casey2017	0.589	0.427	0.263	0.437
kerguelen2014	0.550	0.348	0.509	0.470
kerguelen2015	0.533	0.461	0.597	0.569

5. CONCLUSION

We presented NAVE, a single Conformer detector for Antarctic blue and fin whale calls built on a normalised, adaptive front-end. Under leave-one-site-out cross-validation the single model is ahead of both published pipelines for this task, including the stronger WhaleVAD-BPN, and it reaches that operating point from one forward pass and a per-class threshold, without a boundary-proposal stage or a post-processing search. Its advantage comes chiefly from the class the staged pipelines detects least well, the D downsweep, which the normalised front-end recovers from the background, so that this gain, together with a gain on BP, outweighs a modest deficit on the easier BMABZ class that the staged pipelines detect best. A single model thus advances the hardest class on this task while leading overall.

Two threads frame the result. The development evidence points to D lying in the front-end rather than the detector head: only the normalised channel moved it, while losses, a temporal decoder, and wider context did not. We read its main limitation as feature sensitivity and the fixed training audio rather than capacity, so the clearest next step is richer or learned features and more labelled sites rather than a larger detector. Separately, the duration-matched depthwise kernel is a consistent lever on aggregate F1, with per-class optima tracking call length, a relationship we leave to confirm across more seeds.

6. ACKNOWLEDGMENT

I thank Paul Primus, Florian Schmid, and Tobias Morocutti, whose input and discussions also helped shape this independent work, and Christiaan Geldenhuys for his helpful correspondence and for releasing the WhaleVAD system, which was a source of inspiration for the Conformer developed here. The experiments were run on the compute cluster of the Institute of Computational Perception at Johannes Kepler University Linz.

7. REFERENCES

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

[2] ——, "Balaenoptera physalus," The IUCN Red List of Threatened Species, 2018.

[3] 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.

[4] L. Jean-Labadye *et al.*, "BioDCASE 2025 task 2: Development set," Zenodo, 2025, version 1 (Erratum).

[5] B. S. Miller, 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.

[6] E. Schall, I. I. Kaya, E. Debusschere, P. Devos, and C. Parcerisas, "Deep learning in marine bioacoustics: A benchmark for baleen whale detection," *Remote Sensing in Ecology and Conservation*, vol. 10, no. 5, pp. 642–654, 2024.

[7] B. S. Miller *et al.*, "An annotated library of underwater acoustic recordings for testing and training automated algorithms for detecting Antarctic blue and fin whale sounds," Australian Antarctic Data Centre, 2020, version 1.

[8] C. Geldenhuys, G. Tonitz, and T. Niesler, "Whale-vad: Whale vocalisation activity detection," in *Proceedings of the 10th Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE 2025)*, Barcelona, Spain, October 2025, pp. 165–169.

[9] C. M. Geldenhuys, G. Tonitz, and T. R. Niesler, "Whalevad-bpn: Improving baleen whale call detection with boundary proposal networks and post-processing optimisation," 2025. [Online]. Available: https://arxiv.org/abs/2510.21280

[10] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017, pp. 1195–1204.

[11] S. Cornell, J. Ebbers, C. Douwes, I. Martín-Morató, M. Harju, A. Mesaros, and R. Serizel, "DCASE 2024 task 4: Sound event detection with heterogeneous data and missing labels," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2024 Workshop (DCASE2024)*, 2024.

[12] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech*, 2020, pp. 5036–5040.

[13] H. Nam, S.-H. Kim, B.-Y. Ko, and Y.-H. Park, "Frequency dynamic convolution: Frequency-adaptive pattern recognition for sound event detection," in *Proc. Interspeech*, 2022, pp. 2763–2767.

[14] V. Lostanlen, J. Salamon, M. Cartwright, B. McFee, A. Farnsworth, S. Kelling, and J. P. Bello, "Per-channel energy normalization: Why and how," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 39–43, 2019.

[15] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 29, 2016.

[16] Y. Wang, P. Getreuer, T. Hughes, R. F. Lyon, and R. A. Saurous, "Trainable frontend for robust and far-field keyword spotting," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5670–5674.

[17] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, "RoFormer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, p. 127063, 2024.

[18] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," in *International Conference on Learning Representations (ICLR)*, 2020.

[19] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, "Averaging weights leads to wider optima and better generalization," in *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

[20] N. Zeghidour, O. Teboul, F. de Chaumont Quitry, and M. Tagliasacchi, "LEAF: A learnable frontend for audio classification," in *International Conference on Learning Representations (ICLR)*, 2021.

[21] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, and L. Zelnik-Manor, "Asymmetric loss for multi-label classification," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 82–91.

[22] N. Abraham and N. M. Khan, "A novel focal Tversky loss function with improved attention U-Net for lesion segmentation," in *IEEE International Symposium on Biomedical Imaging (ISBI)*, 2019, pp. 683–687.