ESO: An evolutionary algorithm for efficient model reduction via automated frequency band selection in passive acoustic monitoring - Supplementary results

Lorène Jeantet[1,2,3†], Ufuk Çakır[*4], Joel Lontsi[5], Albert Agisha[6], and Emmanuel Dufourq[1,2,3]

[1]African Institute for Mathematical Sciences, 6 Melrose Road, Muizenberg, 7945, Cape Town, South Africa

[2]African Institute for Mathematical Sciences Research and Innovation Centre, KG 590 St, Kigali, Rwanda

[3]Department of Mathematical Sciences, Stellenbosch University, Victoria Street, Stellenbosch, South Africa

[4]Interdisciplinary Center for Scientific Computing, Heidelberg University, Germany

[5]Zindi, South Africa

[6]Fakultät Elektronik und Informatik, Aalen University of Applied Sciences, Anton-Huber-Straße 25, 73430 Aalen, Germany

[†]Corresponding author. Email: lorene@aims.ac.za

[*] Now at Intelligent Earth Centre for Doctoral Training, University of Oxford, Oxford, UK

1 Materials and Methods

1.1 Datasets

To assess the efficacy of an evolutionary algorithm as a valuable tool for automatically identifying regions of interest within spectrograms for acoustic classification while simultaneously minimising computational costs, we tested ESO on three datasets obtained through PAM of three different species. These datasets were previously described in Dufourq, Batist, Foquet, and Durbach (2022) and are available online. The Hainan gibbon dataset comprises 76 hours of acoustic files recorded at 9 600 Hz using eight SongMeter SM3 recorders (Wildlife Acoustics, Maynard, Massachusetts) deployed in Bawangling Division, Hainan Tropical Rainforest National Park Bureau, in Hainan, China, between March and August 2016 (Dufourq et al., 2021). The Hainan gibbon is classified as critically endangered on the International Union for Conservation of Nature's Red List of Threatened Species (IUCN Red list) with only a single population surviving in Hainan (Geissmann, T. and Bleisch, W., 2020; Liu, Ma, Cheyne, & Turvey, 2020). The Thyolo Alethe dataset comprises 26 hours of acoustic files recorded at 32 000 Hz using 10 AudioMoths (Hill, Prince, Snaddon, Doncaster, & Rogers, 2019) in the Mount Mulanje Biosphere Reserve, Malawi, during five days in November 2020. The Thyolo Alethe is classified as a vulnerable species on the IUCN Red list and is endemic to Malawi and Mozambique (BirdLife International, 2018a). The Pin-tailed Whydah dataset comprises 58 hours of acoustic files recorded at 48 000 Hz using one AudioMoth at the Intaka Island Nature Reserve in Cape Town, South Africa over two weeks in January 2021. The Pin-tailed Whydah is classified as a species of least concern and can be found in all non-arid regions of Africa (BirdLife International, 2018b).

For each species, the acoustic data were analysed by simultaneously listening to and visualising the corresponding spectrograms (Dufourq et al., 2022, 2021). The start and end times of the target species' calls were manually annotated. Additionally, windows of the soundscape containing biophony, anthropophony, and geophony, in the absence of the target species' calls, were annotated as examples of the negative class. For these three datasets, the aim was to develop a binary classifier that could automatically detect the presence or absence of the target species. ESO was tasked with finding the smallest set of informative frequency bands, with an emphasis on minimizing their number. Its performance was compared to that of a baseline model trained on the full spectrogram, as is typically done in most deep learning applications in PAM (Batist et al., 2024; Dufourq et al., 2021;

Stowell, 2022). To train the baseline model, execute ESO, and compare the two approaches, each dataset was randomly split into training, validation, and test sets prior to preprocessing (Table 1). Following common conventions in the literature, 60% of the recorded files were randomly selected for the training set, 20% for the validation set, and 20% for the test set. This partitioning ensures that the model is evaluated on segments from recordings it has not previously encountered.

To balance the training datasets (presence and absence classes), we used data augmentation to increase the size of the minority class, or randomly removed segments from the majority class. Data augmentation involved three methods, namely, time shifting, blending, and adding noise (Jeantet & Dufourq, 2023). Time shifting entailed randomly selecting a time point within the segment and shifting the segment's start to that point, wrapping the segment to maintain its original duration. Blending involved randomly selecting two segments—one from the minority class and another from the negative class—and combining them using the formula $\alpha \times x_{s1} + (1 - \alpha) \times x_{s2}$, where x_{s1} and x_{s2} are the two randomly selected segments, and α is a weighting factor ($alpha = 0.2$ in this study). To introduce noise into a segment, random samples were generated from a normal distribution (mean of zero, standard deviation of one) and added to the original segment, scaled by a factor of 0.0009. All three augmentation methods were applied proportionally to increase the size of the minority class to match that of the majority class.

Table 1: Summary of the acoustic dataset for the three species. Each dataset was framed as a binary classification problem (presence/absence of the species' vocalisation).

Metric	**Hainan gibbon**	**Thyolo Alethe**	**Pin-tailed Whydah**
Dataset Size			
Total number of audio files	69	68	174
Total duration (hours)	76	26	58
Training Set (before augmentation)			
Presence/absence segments	5 373 / 6 541	7 564 / 3 631	3 877 / 5 596
Training Set (after augmentation)			
Presence/absence segments	6 543 / 6 541	7 564 / 7 567	5 599 / 5 596
Validation Set			
Presence/absence segments	2 347 / 1 632	1 498 / 2 372	1 418 / 1 852
Test Set			
Number of test audio files	13	12	33
Test duration (hours)	11	5	11

1.2 ESO: Evolutionary Spectrogram Optimisation

1.2.1 Genetic operations

A genetic algorithm optimises a problem by evolving its population, mirroring Darwinian natural selection, where individuals with the highest fitness have a greater chance of reproducing and passing on favourable traits that enhance survival. Similarly, the evolution of the population in ESO relies on the selection of candidate solutions with the highest fitness – known as parent selection – which are then used to generate new individuals (offspring), through nature-inspired genetic operations such as reproduction, mutation, and crossover.

Parent selection – Each genetic operation starts with the selection of parents based on their fitness scores. Although there are various methods for parent selection (Blickle & Thiele, 1996), ESO uses tournament selection. Tournament selection is based on a user-defined parameter k, which determines the size of the subset from which the parents are chosen. Therefore, tournament selection starts by randomly selecting a subset of k chromosomes from the population. These k chromosomes are compared with each other and the one with the best fitness is kept. For each run of tournament selection, a single parent is selected. Hence if n individuals are required, the tournament selection algorithm is executed n times.

Reproduction – Reproduction is an asexual operator that only requires one parent to generate one offspring. Tournament selection is used to obtain one parent which is copied from the current generation to the next one.

Mutation – Mutation is an asexual operator and requires one parent using tournament selection. It creates a new chromosome by adding random variation to one of the genes inside the selected parent chromosome (Figure 1, A). The choice of gene to mutate is random. Since a gene encodes the position and height of a band, there are three possible options for performing the mutation. For the first option, a small random value is added to the band position ($P_t' = P_t + \delta P$). The second option changes the band height ($h_k' = h_k + \delta h$). The third option combines the previous two options, by mutating the position and height of the band. During execution, one of these three options is randomly selected. The offspring is added to the new population.

Crossover – Crossover is a sexual operator, it requires two parents (using tournament selection twice) and creates two new chromosomes by exchanging genes between the parents (Figure 1, B). If the parents have different numbers of genes, their lengths are evaluated, and the shorter chromosome is identified. A crossover point is then randomly chosen within the length of the shorter chromosome. Beyond this point, genes between the two parents are swapped, resulting in two offspring. The offspring are added to the new population.

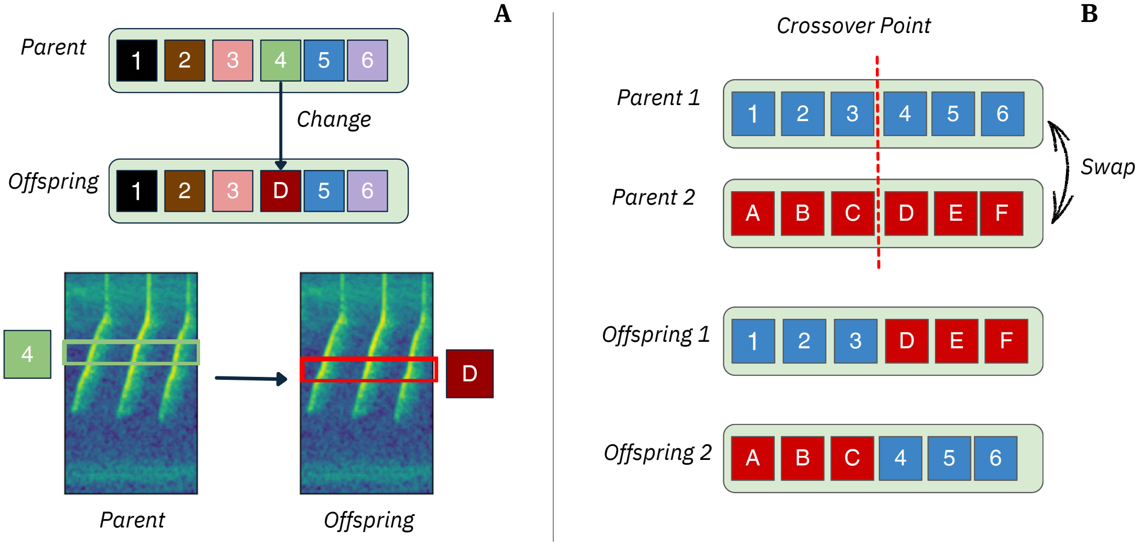

Figure 1: Representation of the modifications induced by the genetic operations, mutation (A), and crossover (B), used in the ESO algorithm to generate new individuals.

The population in each generation is created by applying the genetic operators to the individuals from the previous generation, while maintaining a constant population size across generations. Parents are always selected from the previous generation, with replacement, thus a parent can be selected more than once. The only exception is the initial population, which is randomly created. The genetic operators are applied using different percentages (10% reproduction, 30% mutation, and 60% crossover). These percentages are user-defined parameters and can be adjusted for each use of ESO.

1.3 Performance evaluation of the baseline and ESO models

To compare the efficiency of the baseline and ESO chromosome models in a scenario closest to real-world application, we applied them on the entire audio files of the testing dataset. A sliding window approach was used from the start to the end of each file, maintaining the same segment duration as in the preprocessing step for each species, with a one-second overlap. Each window was low-pass filtered and downsampled before being converted into spectrograms for the baseline model. For the ESO chromosome model, the windows were directly processed into spectrograms, and the frequency bands encoded by the ESO chromosome were extracted. The model was then applied to each spectrogram, producing probability estimates for both the presence and absence of the target species' calls. A positive prediction was assigned only if the associated model probability exceeded 0.8. Consecutive positively predicted spectrograms were then grouped, with a sequence considered valid only if at least three consecutive windows were classified as positive. Isolated positively predicted spectrograms that were not part of a sequence of at least three were disregarded. This criterion did not apply to the Thyolo Alethe dataset, which is based on short segment durations (one second), and all positively predicted spectrograms were preserved. The calling bout was reconstructed by defining the start time as the beginning of the first positively predicted spectrogram in the sequence and the end time as the last positively predicted spectrogram's endpoint.

We compared the predicted calling bouts to the manually annotated calls, with a calling bout considered a TP if it overlapped with an annotated call by more than 25% of the segment duration of the target species. For the Thyolo Alethe dataset, this threshold was lowered to 10% due to the species' more challenging call detection. Call bouts were classified as FP if they had no overlap exceeding 25% with any manually annotated call. Manually annotated calls with no overlapping predicted bouts were considered FN. The number of TN was calculated by generating non-overlapping windows of segment duration outside the manually annotated calls, representing the absence of the species' calls. TN was defined as the number of these windows that did not overlap with predicted calling bouts. Using the TP, TN, FP, and FN values obtained, the F1-score was computed as described in Section 2.3.

To assess the model's suitability for deployment on low-resource devices, we calculated additional metrics related to energy consumption and RAM usage for each model. To evaluate and compare the computational cost, we computed the floating point operations (FLOPs), which estimates the number of arithmetic operations (e.g., additions and multiplications) performed on floating-point numbers during the inference of a single spectrogram. We obtained these values using the fvcore library (`https://github.com/facebookresearch/fvcore`) in Python. To assess RAM usage during inference, we used the psutil library (`https://github.com/giampaolo/psutil`) in Python to monitor memory consumption. We first cleared unused memory and removed residual data linked between objects. Then, we ran a function that loads WAV files, preprocesses them only for the baseline model, applies a sliding window, generates spectrograms, and predicts the target species' call. Throughout this process, RAM usage was recorded every 0.1 seconds. We then calculated the minimum, maximum, and mean memory usage for the entire process. To limit interference, Wi-Fi was disabled and no other applications were run. Finally, we estimated the energy consumption of the CPU, GPU and RAM throughout the process using the CodeCarbon library (Courty et al., 2024).

2 Results

Table 2: Comparison of spectrogram sizes and computational performance of a CNN trained on ESO-optimised vs. downsampled and low-pass filtered spectrograms across three datasets. Inference time refers to the total duration needed for data loading, preprocessing for the baseline model, prediction, and calling bout reconstruction across the test set. FLOPs indicate the number of floating-point operations required for the inference of a single spectrogram.

Metric	**Hainan gibbon**	**Thyolo Alethe**	**Pin-tailed Whydah**
Spectrogram Size			
Baseline	(128 x 76)	(128 x 26)	(128 x 144)
ESO (Best Chromosome)	(19 x 151)	(13 x 126)	(42 x 188)
Difference	-70.51%	-50.78%	-57.16%
Number of Genes	3	1	6
F1-score			
Baseline	89.02	88.45	74.82
ESO (Best Chromosome)	**94.64**	**90.04**	**79.48**
Difference	+6.65%	+1.8%	+6.23%
Number of Parameters			
Baseline	132 234	32 394	262 794
ESO (Best Chromosome)	**29 322**	**9 098**	**93 834**
Difference	-77.82%	-71.91%	-64.29%
Inference Time (s)			
Baseline	183	120	347
ESO (Best Chromosome)	**168**	**68**	**189**
Difference	-18.75%	-43.33%	-45.51%
FLOPs			
Baseline	4 406 336	1 208 896	8 749 632
ESO(Best Chromosome)	913 472	374 080	3 244 096
Difference	-79.20%	-69.06%	-62.92%
Size on Disk (kB)			
Baseline	520	132	1 000
ESO (Best Chromosome)	120	40	372
Difference	-76.92%	-69.70%	-62.80%

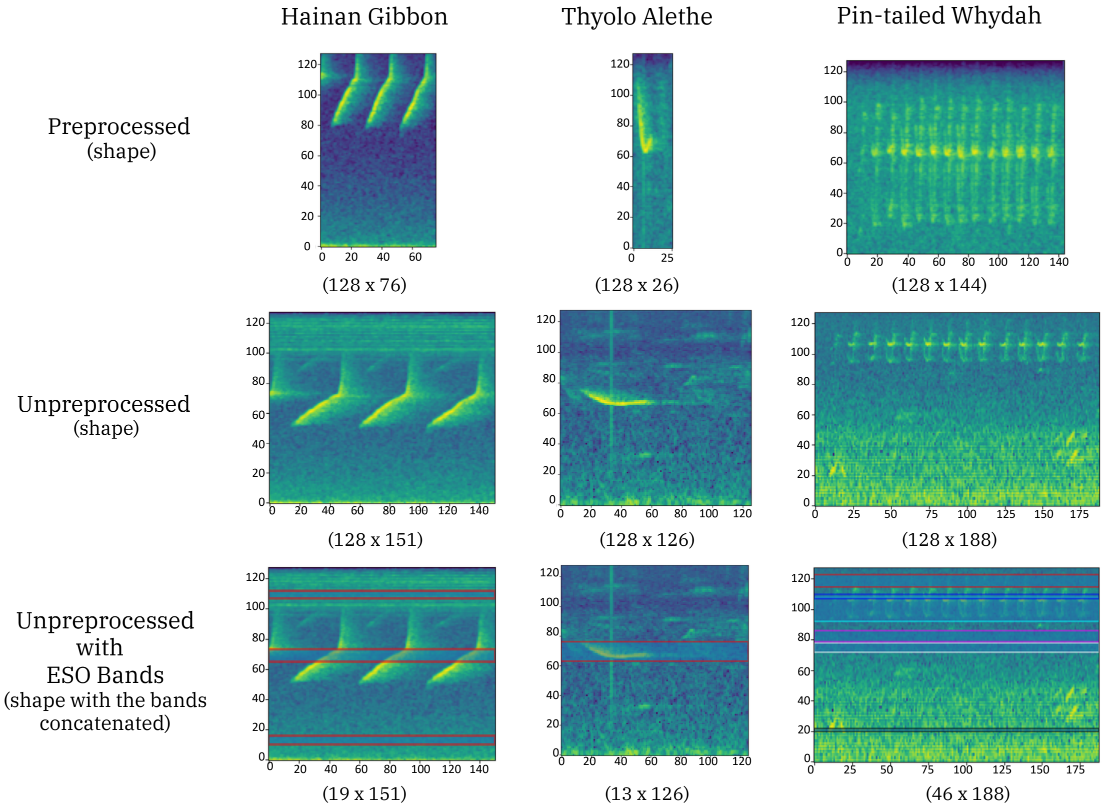

Figure 2: Spectrogram representations for the three species. The first row shows preprocessed spectrograms used to train the baseline model, the second row presents unprocessed spectrograms (without filtering and downsampling) for the ESO algorithm, and the third row displays spectrograms with frequency bands selected by the best chromosome of the ESO algorithm. We used different colours to represent the frequency bands extracted for the Pin-tailed Whydah to distinguish overlapping bands. The vertical axis indicates frequency, while the horizontal axis represents time.

Table 3: Comparison of RAM usage and energy consumption during the automated detection of species calls in the test dataset across the three datasets. Measurements cover inference operations such as data loading, downsampling and filtering in the baseline model, and segmentation, spectrogram generation, frequency band extraction, and output reconstruction in the ESO best chromosome model.

Metric	**Hainan gibbon**	**Thyolo Alethe**	**Pin-tailed Whydah**
Max RAM Usage (MB)			
Baseline	2 570.8	2 189.1	2 289.6
ESO Best Chromosome	1 930.8	1 207.2	1 206.0
Difference	-24.89%	-44.85%	-47.33%
Mean RAM Usage (MB)			
Baseline	1 392.7	1 290.0	1 490.1
ESO Best Chromosome	1 180.0	1 074.4	1 124.4
Difference	-15.27%	-16.71%	-24.54%
RAM Energy Consumed (Wh)			
Baseline	0.296	0.186	0.519
ESO Best Chromosome	0.254	0.125	0.239
Difference	-14.19%	-32.80%	-53.95%
CPU Energy Consumed (Wh)			
Baseline	2.139	1.345	3.750
ESO Best Chromosome	1.839	0.909	1.725
Difference	-14.02%	-32.42%	-54.00%
GPU Energy Consumed (Wh)			
Baseline	0.264	0.113	0.856
ESO Best Chromosome	0.166	0.075	0.273
Difference	-37.12%	-33.63%	-68.11%
Total Energy Consumed (Wh)			
Baseline	2.699	1.643	5.124
ESO Best Chromosome	2.259	1.110	2.236
Difference	-16.30%	-32.44%	-56.36%

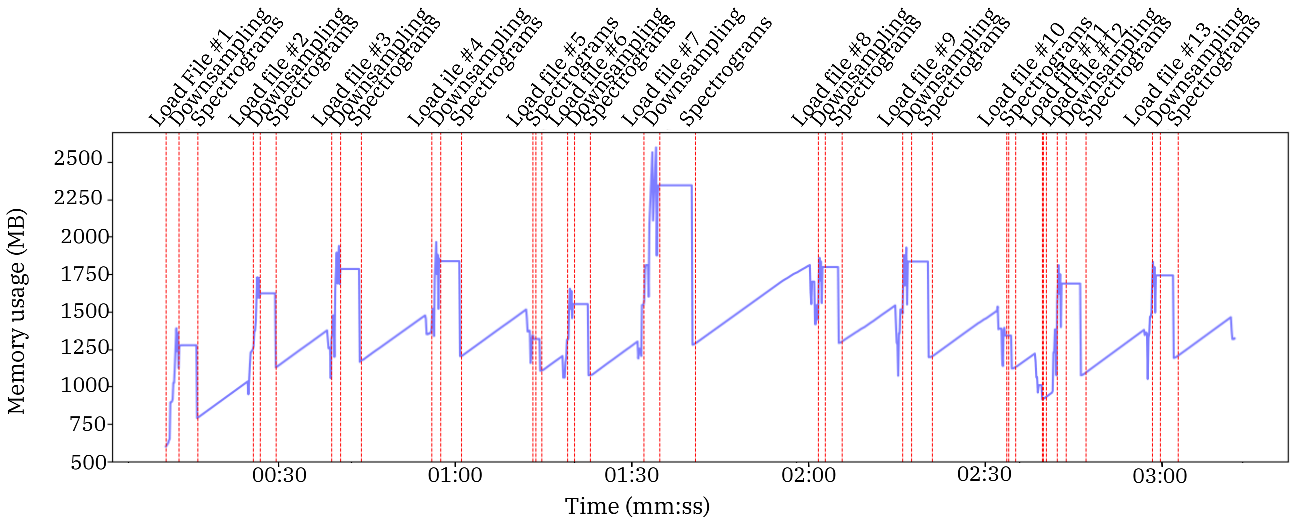

Figure 3: Evolution of memory usage during inference of the test dataset with downsampling and low-filtering of the audio files before their conversion into spectrogram. The term ”Spectrograms” in the figure indicates the beginning of the conversion of the extracted audio windows into spectrograms.

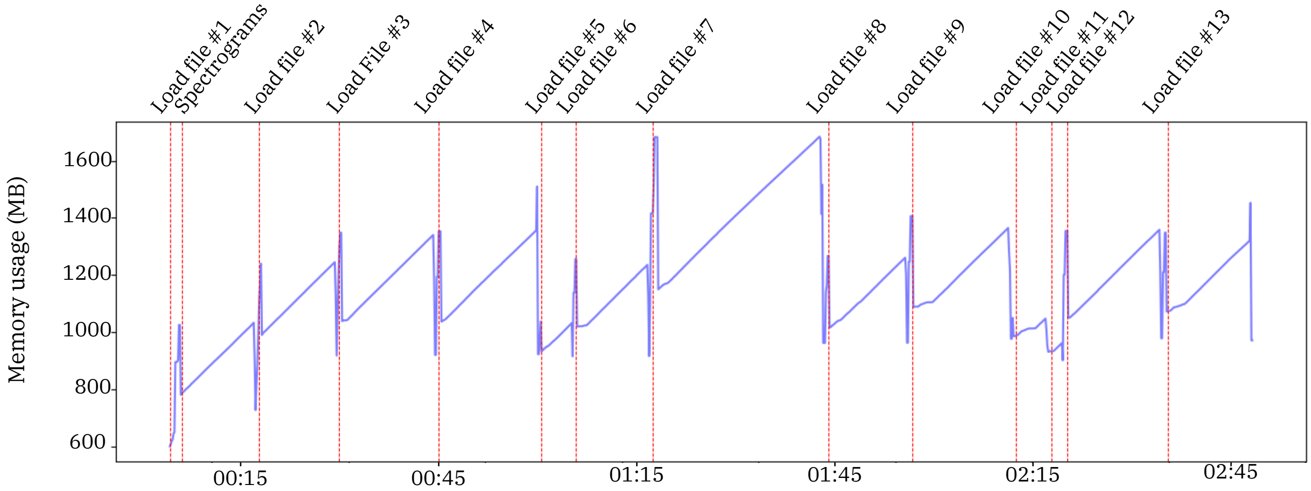

Figure 4: Evolution of memory usage during inference of the test dataset without downsampling and low-filtering of the audio files and with extraction of the frequency bands identified by the ESO algorithm. The term ”Spectrograms” in the figure indicates the beginning of the conversion of the extracted audio windows into spectrograms.

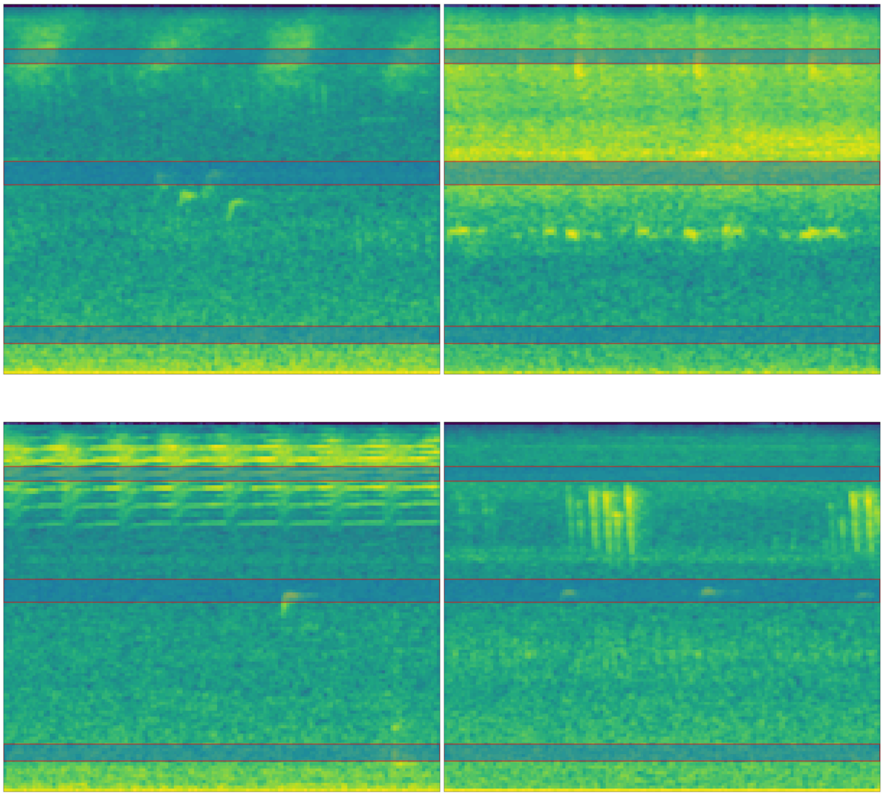

Figure 5: Example of false positive spectrograms from the baseline model that were correctly classified as negative windows by the ESO best chromosome model. The red boxes indicate the extracted frequency bands.

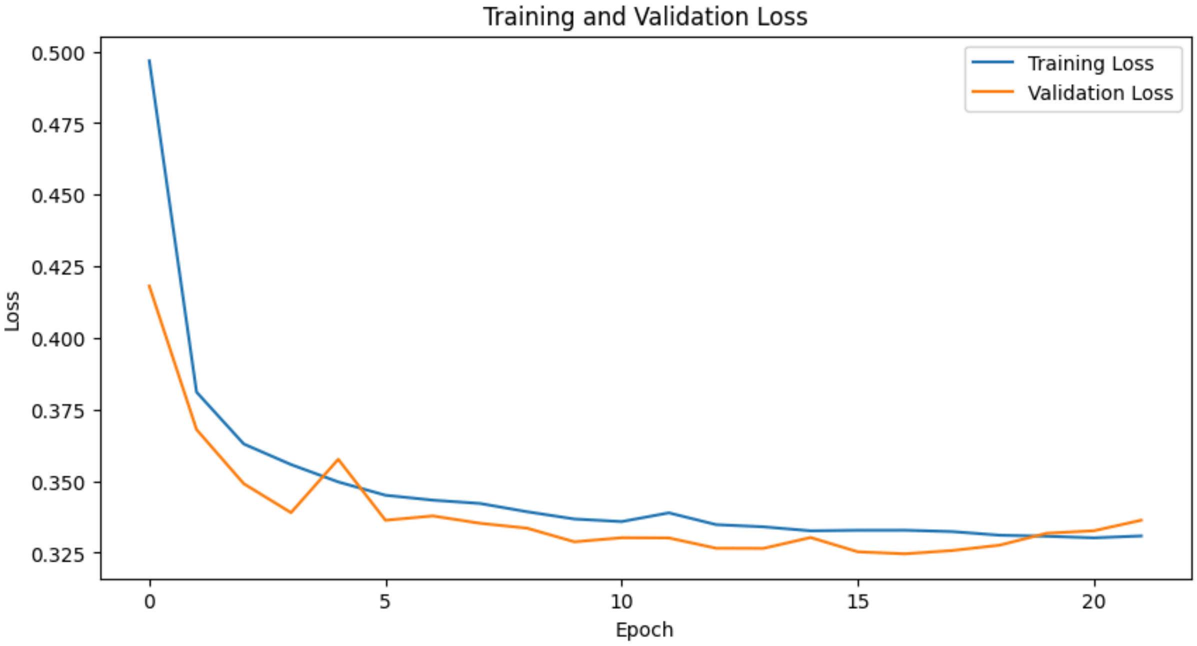

Figure 6: Plot of the loss for the training and validation Hainan gibbon datasets with the baseline model trained with learning rate of 0.001.

Hainan gibbon

Baseline model

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	653	109
Predicted negative (0)	52	9 214

ESO best chromosome

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	662	32
Predicted negative (0)	43	9 396

Thyolo alethe

Baseline model

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	1 957	297
Predicted negative (0)	214	17 774

ESO best chromosome

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	1 912	164
Predicted negative (0)	259	18 018

Pin-tailed whydah

Baseline model

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	327	147
Predicted negative (0)	73	18 486

ESO best chromosome

	Presence of call (1)	Absence of Call (0)
Predicted positive (1)	335	108
Predicted negative (0)	65	18 610

Figure 7: Confusion matrices obtained with the baseline model and the ESO best chromosome for the three datasets.

References

Batist, C. H., Dufourq, E., Jeantet, L., Razafindraibe, M. N., Randriamanantena, F., & Baden, A. L. (2024). An integrated passive acoustic monitoring and deep learning pipeline for black-and-white ruffed lemurs (varecia variegata) in ranomafana national park, madagascar. *American Journal of Primatology*, 1–18. doi: 10.1002/ajp.23599

BirdLife International. (2018a). Chamaetylas choloensis. *The IUCN Red List of Threatened Species 2018*. (Accessed on 19 May 2025) doi: 10.2305/IUCN.UK.2018-2.RLTS.T22709004A131333396.en

BirdLife International. (2018b). Vidua macroura. *The IUCN Red List of Threatened Species 2018*. (Accessed on 19 May 2025) doi: 10.2305/IUCN.UK.2018-2.RLTS.T22709004A131333396.en

Blickle, T., & Thiele, L. (1996). A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation*, *4*(4), 361–394. doi: 10.1162/evco.1996.4.4.361

Courty, B., Schmidt, V., Goyal-Kamal, MarionCoutarel, Feld, B., Lecourt, J., ... MinervaBooks (2024). *mlco2/codecarbon: v2.4.1.* Zenodo. doi: 10.5281/zenodo.11171501

Dufourq, E., Batist, C., Foquet, R., & Durbach, I. (2022). Passive acoustic monitoring of animal populations with transfer learning. *Ecological Informatics*, *70*, 101688. doi: 10.1016/j.ecoinf.2022.101688

Dufourq, E., Durbach, I., Hansford, J. P., Hoepfner, A., Ma, H., Bryant, J. V., ... Turvey, S. T. (2021). Automated detection of Hainan gibbon calls for passive acoustic monitoring. *Remote Sensing in Ecology and Conservation*, *7*(3), 475–487. doi: 10.1002/rse2.201

Geissmann, T. and Bleisch, W. (2020). Nomascus hainanus. *The IUCN Red List of Threatened Species 2020*. (Accessed on 21 November 2024) doi: 10.2305/IUCN.UK.2020-2.RLTS.T41643A17969392.en

Hill, A. P., Prince, P., Snaddon, J. L., Doncaster, C. P., & Rogers, A. (2019). AudioMoth: A low-cost acoustic device for monitoring biodiversity and the environment. *HardwareX*, *6*, e00073. doi: 10.1016/j.ohx.2019.e00073

Jeantet, L., & Dufourq, E. (2023). Improving deep learning acoustic classifiers with contextual information for wildlife monitoring. *Ecological Informatics*, *77*, 102256. doi: 10.1016/j.ecoinf.2023.102256

Liu, H., Ma, H., Cheyne, S. M., & Turvey, S. T. (2020). Recovery hopes for the world's rarest primate. *Science*, *368*(6495), 1074–1074. doi: 10.1126/science.abc1402

Stowell, D. (2022, March). Computational bioacoustics with deep learning: a review and roadmap. *PeerJ*, *10*, e13152. doi: 10.7717/peerj.13152