Université du Québec
Institut national de la recherche scientifique
Centre Énergie Matériaux Télécommunications

Domain-Enriched Knowledge Distillation for Robust Audio Edge Applications

Distillation de connaissances enrichie par des informations de domaine pour des applications audio robustes en périphérie de réseau

Par

Heitor Rodrigues Guimarães

Thèse présentée pour l'obtention du grade de
Doctorat en sciences, Ph.D.
en Télécommunications

Jury d'évaluation

Président du jury et examinateur interne	Prof. Douglas O'Shaughnessy INRS-EMT
Examinateur externe 1	Prof. Alessandro Koerich École de Technologie Supérieure
Examinateur externe 2	Prof. Mirco Ravanelli Concordia University / Mila Quebec AI Institute
Directeur de recherche	Prof. Tiago H. Falk INRS-EMT
Codirecteur de recherche	Prof. Anderson R. Avila INRS-EMT

©Heitor Rodrigues Guimarães, 2026

Acknowledgements

This work would not have been possible without the support and guidance of countless people throughout this journey. First, my deepest gratitude to my supervisor, Prof. Tiago Falk. Your dedication and empathy have always been a source of inspiration. I am deeply grateful for your trust, encouragement, and for giving me this opportunity. I would also like to thank Prof. Anderson for his unwavering support and collaboration throughout my doctoral studies.

I am grateful to have met so many incredible people during this journey, especially my MuSAE friends: Arthur Pimentel, Marília Soares, Belmir Jesus, Yi Zhu, Mahsa Abdollahi, Abhishek Tiwari, Mina Jaberi, and Ashkan Moradi. Thank you for the discussions, the friendship, and the shared moments that made this journey lighter and more amusing.

During this journey, I was lucky to meet and collaborate with so many incredible people during internships. I am grateful to everyone that I met during this period, especially my hosts Jiaqi Su (Adobe), Zeyu Jin (Adobe), Ke Tan (Meta), and Buye Xu (Meta), for their support and for showing me how much fun (and challenging) it is doing research in industry.

Finally, I would like to thank my family and friends. I am deeply grateful to my parents and grandparents for their unconditional love, constant encouragement, and for always celebrating even the smallest victories. Last but not least, to my wife, Jessica Guimarães: thank you for choosing to embark on this adventure with me right after we got married. I deeply appreciate all the sacrifices you made so that we could be here together.

Abstract

Recent advances in the deep learning field, particularly the shift toward self-supervised learning (SSL), have revolutionized the landscape of audio and speech processing by enabling the creation of generalist foundation models capable of learning meaningful representations from unlabeled data. While these models offer a promising unified backbone for diverse applications, potentially reducing computational redundancy and enhancing data privacy, their massive scale and parameter count present a critical bottleneck for deployment on resource-constrained edge devices. This deployment challenge is compounded by a significant trade-off: techniques required to compress these models for the edge often make them disproportionately vulnerable to "in-the-wild" conditions. Therefore, we aim to mitigate the trade-off between efficiency and robustness, specifically by addressing the susceptibility of compressed encoders to environmental distribution shifts and so-called adversarial attacks. To this end, in this doctoral thesis we propose several methodological innovations divided into two primary application domains: speech and bioacoustics monitoring.

First, we propose a task-agnostic knowledge distillation framework that jointly compresses speech encoders and enhances their robustness against environmental perturbation (e.g., additive noise and reverberation). By enforcing feature denoising and signal reconstruction objectives during distillation, the student model is forced to learn noise-invariant representations. We evaluate our compressed encoder across 12 downstream from the SUPERB benchmark, demonstrating that our methodology yields significant gains in out-of-domain and in-domain scenarios.

Second, we address security vulnerabilities by proposing a robust distillation method designed to compress speech encoders while simultaneously fortifying them against L_p-norm adversarial attacks. Focusing on keyword spotting systems, we demonstrate that imposing geometric priors (variance, invariance, and covariance regularization) on the latent representations of the student model yields significantly more robust architectures. Empirical results on the Google Speech Commands dataset indicate that this technique outperforms current state-of-the-art robust distillation methods.

Finally, transitioning to the bioacoustics domain, we first establish a rigorous benchmark for IoT-enabled acoustic beehive monitoring. We evaluate a suite of general-purpose and specialized audio encoders across four critical colony management tasks, demonstrating that foundation models possess strong generalizability even with limited labeled data. Next, we propose a family of resource-efficient audio encoders designed specifically for ecological monitoring. Our main contribution in this work is the integration of a novel DSP-inspired inductive bias: modulation-aware acoustic features via the so-called FiLM conditioning. This design exploits two properties of bioacoustic signals: (i) signal and environmental noise, while often overlapping in time and frequency, are separable in the modulation domain; and (ii) the periodic nature of animal vocalizations. Experimental results show that our architecture achieves state-of-the-art performance across 10 diverse bioacoustic tasks

(plus our beehive benchmark). Notably, our compact models match or exceed the performance of substantially larger baselines, confirming their suitability for real-world IoT deployments.

Keywords Self-Supervised Learning, Foundation Models, Knowledge Distillation, Distribution Shifts, Adversarial Attacks, Speech Processing, Bioacoustic Applications, Internet-of-Things (IoT)

Résumé

Les avancées récentes dans le domaine de l'apprentissage profond (*Deep Learning*), et plus particulièrement la transition vers l'apprentissage auto-supervisé (*Self-Supervised Learning*, SSL), ont révolutionné le traitement de l'audio et de la parole en permettant la création de modèles de fondation généralistes, capables d'apprendre des représentations significatives à partir de données non étiquetées. Bien que ces modèles offrent une ossature unifiée prometteuse pour diverses applications, réduisant potentiellement la redondance computationnelle et améliorant la confidentialité des données, leur taille massive et leur nombre élevé de paramètres constituent un goulot d'étranglement critique pour leur déploiement sur des dispositifs en périphérie de réseau (*edge devices*), limités en ressources. Ce défi de déploiement est exacerbé par un compromis majeur: les techniques nécessaires pour compresser ces modèles en vue de leur utilisation en périphérie les rendent souvent disproportionnellement vulnérables aux conditions "réelles" (*in-the-wild*). Par conséquent, cette thèse vise à atténuer ce compromis entre efficacité et robustesse, en s'attaquant spécifiquement à la sensibilité des encodeurs compressés face aux dérives de distribution environnementales et aux attaques adverses. À cette fin, nous proposons plusieurs innovations méthodologiques réparties dans deux domaines principaux: le traitement de la parole et la surveillance bioacoustique.

Premièrement, nous proposons un cadre de distillation de connaissances agnostique à la tâche, qui compresse conjointement les encodeurs de parole et améliore leur robustesse face aux perturbations environnementales (par ex., bruit additif et réverbération). En imposant des objectifs de débruitage de caractéristiques et de reconstruction du signal lors de la distillation, le modèle étudiant est contraint d'apprendre des représentations invariantes au bruit. Nous évaluons notre encodeur compressé sur 12 tâches en aval du banc d'essai SUPERB, démontrant que notre méthodologie engendre des gains significatifs tant dans des scénarios hors domaine que dans le domaine cible.

Deuxièmement, nous abordons les vulnérabilités de sécurité en proposant une méthode de distillation robuste conçue pour compresser les encodeurs de parole tout en les fortifiant simultanément contre les attaques adverses bornées par la norme L_p. En nous concentrant sur les systèmes de détection de mots-clés (*Keyword Spotting*), nous démontrons que l'imposition de priors géométriques (régularisation de variance, invariance et covariance) sur les représentations latentes du modèle étudiant produit des architectures significativement plus robustes. Les résultats empiriques sur le jeu de données Google Speech Commands indiquent que cette technique surpasse les méthodes de distillation robuste actuelles de l'état de l'art.

Enfin, en transitionnant vers le domaine bioacoustique, nous établissons d'abord un banc d'essai rigoureux pour la surveillance acoustique des ruches assistée par l'IoT. Nous évaluons une suite d'encodeurs audio généralistes et spécialisés sur quatre tâches critiques de gestion des colonies, démontrant que les modèles de fondation possèdent une forte capacité de généralisation, même avec des données étiquetées limitées. Ensuite, nous proposons une famille d'encodeurs audio

économes en ressources, conçus spécifiquement pour la surveillance écologique. Notre principale contribution réside dans l'intégration d'un biais inductif novateur inspiré du traitement du signal (DSP): des caractéristiques acoustiques sensibles à la modulation via un conditionnement FiLM. Cette conception exploite deux propriétés des signaux bioacoustiques: (i) le fait que le signal et le bruit environnemental, bien que se chevauchant souvent en temps et en fréquence, sont séparables dans le domaine de la modulation ; et (ii) la nature périodique des vocalisations animales. Les résultats expérimentaux montrent que notre architecture atteint des performances de pointe sur 10 tâches bioacoustiques diverses (en plus de notre banc d'essai apicole). Notamment, nos modèles compacts égalent ou dépassent la performance de modèles de référence substantiellement plus grands, confirmant leur adéquation pour des déploiements IoT en conditions réelles.

Mots-clés Apprentissage auto-supervisé, modèles de fondation, distillation de connaissances, dérives de distribution, attaques adverses, traitement de la parole, applications bioacoustiques, Internet des objets (IoT).

Contents

Acknowledgements **iii**

Abstract **v**

Résumé **vii**

Contents **ix**

List of Figures **xiii**

List of Tables **xv**

Synthèse **1**
0.1 Introduction . . . 1
0.1.1 Domaines d'application et implications pratiques . . . 4
0.1.2 Contributions de la thèse . . . 6
0.1.3 Organisation de la thèse . . . 7
0.2 Résumé . . . 8
0.2.1 Chapitre 2: Contexte . . . 8
0.2.2 Chapitre 3: Caractéristiques invariantes au bruit via distillation débruitante et apprentissage multi-tâches . . . 13
0.2.3 Chapitre 4: Caractéristiques robustes aux attaques adverses via l'analyse de corrélation canonique . . . 16
0.2.4 Chapitre 5: Exploiter les modèles de fondation pour la surveillance acoustique des ruches . . . 20
0.2.5 Chapitre 6 : BioME : un encodeur neuronal bioacoustique sensible aux modulations . . . 22

List of Acronyms and Abbreviations **1**

1 Introduction **25**
1.1 Application domains and practical implications . . . 28
1.1.1 Speech processing . . . 28
1.1.2 Bioacoustic monitoring . . . 28
1.2 Thesis contributions . . . 29
1.3 List of publications . . . 30
1.4 Thesis organization . . . 33

2 Background **35**

2.1 Deep Neural Networks . . . 35
2.2 Self-Supervised Learning . . . 37
2.2.1 Speech Encoders . . . 37
2.2.2 General-purpose Audio Encoders . . . 38
2.2.3 Assessing the quality of the representations . . . 39
2.3 Model compression . . . 41
2.3.1 Quantization and Pruning . . . 41
2.3.2 Knowledge Distillation . . . 42
2.4 Domain Adaptation . . . 44
2.5 Adversarial Attacks . . . 45
2.5.1 Formal definition and Adversarial Training . . . 45
2.5.2 Adversarial Robustness in Audio Encoders . . . 46
2.5.3 Robust Knowledge Distillation . . . 46

I Applications to Speech Encoders 49

3 Noise-invariant features via denoising distillation and multi-task learning 51
3.1 Preamble . . . 51
3.2 Introduction . . . 51
3.3 RobustDistiller: Joint distillation and noise-invariant representation extraction . . . 52
3.3.1 Modification #1: Feature Denoising Knowledge Distillation . . . 53
3.3.2 Modification #2: Multi-task learning . . . 54
3.3.3 Modification #3: (Optional) Curriculum learning . . . 55
3.3.4 Notes on extending RobustDistiller to DP-methods . . . 56
3.4 Experimental Setup . . . 57
3.4.1 Datasets . . . 57
3.4.2 Pre-training . . . 57
3.4.3 Downstream Tasks, Testing conditions, and Figures-of-Merit . . . 58
3.5 Experimental Results and Discussion . . . 59
3.5.1 Performance across SUPERB tasks . . . 59
3.5.2 Performance on SUPERB-SG Tasks: Enhancement and Source Separation . . . 65
3.5.3 Speech-noise disentanglement . . . 65
3.5.4 Impact of the Enhancement Head . . . 68
3.5.5 Discussion on distilling from already robust Teachers . . . 69
3.5.6 Effect of noise type in low SNR regimes . . . 69
3.5.7 Effects of room size . . . 70
3.5.8 Ablation on Curriculum Learning experiments . . . 71
3.6 Conclusion . . . 73

4 Adversarially robust features via canonical correlation analysis 75
4.1 Preamble . . . 75
4.2 Introduction . . . 75
4.3 Variance-Invariance-Covariance Knowledge Distillation . . . 76
4.4 Experimental Setup . . . 78
4.5 Experimental Results and Discussion . . . 80
4.5.1 Result #1: Robust accuracy . . . 81
4.5.2 Result #2: Effects of Multi-View inputs . . . 82

4.5.3 Result #3: Extending to more classes . . . 83
4.6 Conclusions . . . 84

II Applications to Bioacoustics with General-Purpose Audio Encoders 85

5 Leveraging Foundation Models for Acoustic Beehive Monitoring 87
5.1 Preamble . . . 87
5.2 Introduction . . . 87
5.3 Selected General-purpose Audio Encoders . . . 90
5.4 Self-Supervised Features for Background Speech Detection in Beehive Recordings . . . 92
5.4.1 Methods and Materials . . . 92
5.4.2 Experimental Results . . . 95
5.5 Benchmarking Audio Representations for General Acoustic Beehive Monitoring . . . 98
5.5.1 Upstream Components: Self-supervised Audio Encoders . . . 99
5.5.2 Featurizer Component: Attending different representations . . . 100
5.5.3 Downstream Components: Datasets, models, and evaluation . . . 100
5.5.4 Summarizing the results with a single score . . . 106
5.5.5 Baselines Models . . . 107
5.5.6 Experimental Results and Discussion . . . 107
5.6 Conclusions . . . 110

6 BioME: A Bioacoustic Modulation-Aware Neural Encoder 113
6.1 Preamble . . . 113
6.2 Introduction . . . 113
6.3 Background and Motivation . . . 114
6.3.1 Bioacoustic Downstream Tasks . . . 114
6.3.2 Modulation Spectrum Signal Representation . . . 115
6.4 Proposed Method: The BioME audio encoder . . . 116
6.4.1 Transformer Encoder . . . 116
6.4.2 Modulation-features via FiLM . . . 118
6.4.3 Layer-wise Knowledge Distillation . . . 119
6.5 Experimental Setup . . . 120
6.5.1 Datasets . . . 120
6.5.2 Hyperparameters and Hardware . . . 120
6.6 Ablation studies . . . 121
6.7 Results on the BEANS benchmark . . . 122
6.8 Acoustic Beehive Monitoring Tasks . . . 124
6.9 Visualizing the impact of MSAB features . . . 126
6.10 Conclusions . . . 129

7 Conclusions and Future Research Directions 131
7.1 Conclusions . . . 131
7.1.1 Speech Processing Applications . . . 132
7.1.2 Bioacoustic Applications . . . 134
7.2 Future Research Directions . . . 135
7.2.1 The self-attention bottleneck . . . 136
7.2.2 Test-time Adaptation for Edge Applications . . . 136

7.2.3 Leveraging synthetic data for audio applications 137

Bibliography **139**

List of Figures

2.1 Schematic overview of the DistilHuBERT framework. The Teacher model remains frozen throughout the process (indicated by the snowflake icon), while the Student model, initialized with the Teacher's weights, is actively optimized during distillation (indicated by the flame icon). 42

2.2 Schematic overview of the Domain Adversarial Training (DAT) framework within Robust HuBERT. A discriminator network, $D(\cdot;\phi)$, processes the latent representations from the context network and is trained to distinguish between clean and noisy inputs. Simultaneously, the encoder is optimized to generate noise-invariant representations designed to fool the discriminator, thereby enforcing robustness against environmental perturbations. 44

3.1 Diagram of the proposed RobustDistiller recipe. The input of the Student model can be a noisy waveform. The Teacher, on the other hand, receives the clean waveform. The Student must learn clean features and simultaneously retrieve as much information as possible to reconstruct a clean utterance. 53

3.2 tSNE visualization on the effect of noise on the latent representation of (top to bottom) HuBERT, Distiller (HuBERT), and RobustDistiller (HuBERT), respectively. Clean signals are shown on the left, while noisy on the right. Data are from eight randomly-selected speakers of LibriSpeech. 67

3.3 Measuring the impact of different enhancement heads and loss functions for our proposed multi-task learning paradigm. 68

3.4 KS accuracy per noise type with clean speech levels used for referencing. 70

3.5 IC performance per room size for different HuBERT distillation schemes. 71

4.1 Diagram of the proposed robust distillation recipe: VIC-KD. 77

4.2 (a) Effects of multiple steps on the PGD-ℓ_∞ method. Next we explore the impact of the hyperparameter ϵ that controls the attack strength in PGD_{40} with (b)ℓ_∞-bounded attacks; and, (c) with ℓ_2-bounded attacks. The step is always calculated as $\alpha = \epsilon/4$. 82

4.3 Effects of the Multi-View on several distillation Methods for the WavLM / TC-ResNet pair. The dashed line represents the robust accuracy of TC-ResNet via TRADES. The Robust Accuracy is measured when the model is attacked by the AutoAttack method. 83

4.4 Scaling robust distillation to a 35-class GSC dataset and comparing it to clean and robust accuracy baselines. AutoAttack is used to generate adversarial perturbations. 84

5.1 Illustrations of a Langstroth hives. In (a), it shows the usual position for the sensor, which is placed over the frames of the brood champer. And (b) the decomposed overall structure. Image reproduced with permission from the authors from [1]. . . . 88

5.2 A 3-minute excerpt from the Nu-hive dataset, from top to bottom, raw audio signal in blue and ground truth labels in green, and spectrogram. 93
5.3 Diagram of the proposed system. (a) is the proposed system built on top of speech foundational models; (b) shows how the same objective is implemented with the BYOL-A upstream, which directly extracts an utterance-level representation on the last layer. 95
5.4 Comparing unsupervised approaches built on top of SSL-based representations against strong baselines. 97
5.5 Schematics of the proposed evaluation system highlighting the three main components: The UpstreamExpert, the Featurizer, and the DownstreamExpert. 99
5.6 Spectrograms of 3-minute excerpts from the NU-Hive dataset. The audios are taken from the same Hive in similar times during the night. (a) is a sample where the queen was present in the hive, and (b) is a sample when the queen was dead. 102
5.7 Data description for the BSTR task: (a) Cumulative density function of labels per set, (b) RMS distribution over the night for two recordings of the same hive. The black line represents the beginning of summer with 20 frames of bees, and the red line represents early fall with half the colony size. 104
5.8 Pre-training experiments with (a) AudioSet-only and (b) AudioSet and the UrBAN 2021 110

6.1 Spectrogram (top row) and the modulation spectrum (bottom row) plots, averaged across samples, for three hummingbird species: Anna's hummingbird (*annhum*), Broad-tailed hummingbird (*brthum*), and Costa's hummingbird (*coshum*). 116
6.2 Block diagram of a single Transformer layer in the proposed BioME encoder. Patch embeddings are processed alongside side-channel context features, which are integrated at each layer through the Conditioner module implementing FiLM-based conditioning. 117
6.3 Modulation Spectrogram Average Bands (MSAB) computation. For example, a 256×256 modulation spectrogram produces a 512-dimensional MSAB feature vector. 118
6.4 Ablation studies for the proposed BioME. We analyze (a) the Student architecture selection, (b) the effect of spectral resolution (NFFT size) on the MSAB features, and (c) model scalability (Edge vs. Small vs. Base). Results are averaged across the classification, detection, and auxiliary tasks of the BEANS benchmark. The 'Avg.' column is the average across all bioacoustic tasks (excluding auxiliary tasks). 121
6.5 Plot showing the relationship between parameter efficiency, scaling properties, and training data partition. 123
6.6 Visualization of a Common Dolphin sample (75003041.wav) from the Watkins dataset. The panels display, from top to bottom: (a) the input mel spectrogram; (b) saliency maps from the ablated BioME model (without modulation features); (c) saliency maps from the full BioME model; and (d) the corresponding modulation spectrogram used as side-channel information (prior to MSAB transformation). 127
6.7 t-SNE visualizations of the latent spaces for BioME (Bio partition, 26M), SS-AST, and BEATs using three hummingbird species (Anna's, Broad-tailed, and Costa's) from the Cornell Bird Identification dataset. To quantify disentanglement, a Decision Tree (max depth=3) was fitted to the 2D projections; the resulting decision boundaries and accuracy scores illustrate the latent disentanglement of each model's representation. 128

List of Tables

3.1 Experimental results for content-related tasks on the SUPERB. **Bold** represents the best result and underline the second best for the compressed models. 60
3.2 Experimental results for speaker-related tasks on the SUPERB. **Bold** represents the best result and underline the second best for the compressed models. 61
3.3 Experimental results for semantics-related tasks on the SUPERB. **Bold** represents the best result and underline the second best for the compressed models. 62
3.4 Experimental results for the paralinguistic task (emotion recognition) on the SUPERB. **Bold** represents the best result and underline the second best for the compressed models. 63
3.5 Overall Score results for each scenario on the original 10 SUPERB tasks grouped by category. **Bold** represents the best result and underline the second best for the compressed models. 65
3.6 Experimental results on the SUPERB-SG benchmark for Speech Enhancement (SE) and Source Separation (SS) tasks. **Bold** represents the best result and underline the second best for the compressed models. 66
3.7 Curriculum Learning: Experimental results for keyword spotting, intent classification, and emotion recognition under clean (**c**), noisy (**n**), reverberation (**r**), and noise-plus-reveberation (**n+r**) test conditions. 72

4.1 Baseline results obtained directly in a supervised setting without knowledge distillation. 78
4.2 Baseline results for the Teacher models in a supervised setting without knowledge distillation. 79
4.3 Baseline results for the Student models in a supervised setting without knowledge distillation. 79
4.4 Results on the GSC dataset with 12 classes for distillation methods. Distilling from standard and robust Teachers. 80

5.1 Experiments for detecting speech sounds on acoustic beehive monitoring, emphasizing our proposed SSL approach. (F) indicates a frozen encoder and (U) unfrozen. 96
5.2 Comparison of related work and our approach. Column "Type" refers to Benchmark or Evaluation studies. "# Models" and "# Tasks" indicate the number of SSL models and tasks evaluated. "Bio?" marks bioacoustic focus, and "ABM" indicates inclusion of at least one Acoustic Beehive Monitoring task. 98
5.3 Detailed description of the data used for each Downstream Task of our system. Columns “Hive Ind.?” and “TVT #samples” describe whether a hive-independent setup is used and the number of samples in the train, validation, and test sets, respectively. 103

5.4 Main evaluation results. **Bold** represents the best and underline the second best average results. 108

6.1 Training datasets and configurations used for our distillation experiments with BioME. 120
6.2 Main evaluation results for the BEANS benchmark. 123
6.3 Results for acoustic beehive monitoring tasks. **Bold** represents the best and underline the second best average results. 125

SYNTHÈSE

Distillation de connaissances enrichie par des informations de domaine pour des applications audio robustes en périphérie de réseau

0.1 Introduction

Les avancées récentes dans le domaine de l'apprentissage profond (deep learning) ont profondément marqué les communautés du traitement de la parole et de l'audio. Elles ont ainsi conduit au développement de systèmes de pointe pour des tâches telles que la reconnaissance automatique de la parole [2, 3], la vérification du locuteur [4], la surveillance de la biodiversité [5, 6], entre autres. En général, ces systèmes reposent sur de grands réseaux neuronaux qui transforment des données d'entrée complexes en représentations plus maniables [7]. Bien que l'apprentissage supervisé ait permis de concevoir des modèles spécialisés excellant dans leurs tâches cibles, cette approche demeure intrinsèquement limitée par la rareté des annotations humaines.

Afin de répondre à cette limitation, la littérature récente s'est concentrée sur le développement de modèles généralistes capables d'exécuter plusieurs tâches et d'acquérir de nouvelles compétences sans dépendre de grandes quantités de données annotées. Dans ce contexte, plutôt que d'apprendre uniquement une correspondance rigide entre l'entrée et la sortie, l'objectif est de concevoir des

modèles prédictifs généralisés qui caractérisent les objets et les signaux en observant le monde, et qui apprennent ainsi à abstraire des concepts de haut niveau (c.-à-d. des caractéristiques ou des représentations).

Apprendre à créer des représentations significatives et désintriquées à partir de signaux de grande dimension, sans supervision explicite, est essentiel dans le cadre de ces travaux. Bien que la caractérisation complète de représentations *significatives* constitue un problème mal défini, certaines propriétés sont souhaitables. Premièrement, les représentations devraient être indépendantes des informations redondantes présentes dans les données afin de capturer efficacement différentes variations [8]. Deuxièmement, les représentations devraient être désintriquées, c'est-à-dire qu'elles séparent les différents facteurs sous-jacents des données, ce qui les rend plus faciles à exploiter dans de multiples tâches en aval [9]. Par exemple, dans le cas des signaux acoustiques, ces caractéristiques apprises capturent des aspects essentiels de la parole et/ou de l'audio, tels que les phonèmes, la représentation dynamique des harmoniques et/ou les transitions spectrales [10].

L'apprentissage auto-supervisé (Self-Supervised Learning, SSL) s'est imposé comme le paradigme dominant pour l'apprentissage de représentations à partir de vastes quantités de données non annotées. Le principe fondamental du SSL consiste à exploiter la structure sous-jacente des données afin de générer des signaux de supervision. Pour entraîner un modèle de fondation[1] capable d'encoder des concepts généralisables, on s'appuie généralement sur la prédiction de parties cachées du signal à partir de contextes observés (par ex., le *masked language modeling* [12]). Cet objectif joue le rôle de *tâche prétexte*, dont le but principal n'est pas la reconstruction en elle-même, mais l'extraction de facteurs explicatifs robustes. Par ailleurs, pour évaluer la qualité de ces représentations apprises, le *linear probing* constitue la méthodologie la plus couramment adoptée. Combinée à des bancs d'essai établis, cette approche fournit un cadre rigoureux et cohérent pour vérifier les performances sur un ensemble diversifié d'applications en aval.

Enfin, les modèles de fondation représentent une alternative prometteuse pour le déploiement en conditions réelles. En servant d'ossature unifiée pour diverses applications, ils éliminent la nécessité d'entraîner et de maintenir des modèles distincts à forte capacité pour chaque tâche spécifique, puisque des représentations partagées sont réutilisées entre applications. Cette réutilisation permet

[1]Tout au long de cette thèse, nous utilisons les termes "modèle SSL", "encodeurs de parole et d'audio" et "modèle de fondation" de manière interchangeable, conformément à la définition de [11]: "Un modèle de fondation est tout modèle entraîné sur des données larges (généralement à l'aide de l'auto-supervision à grande échelle) et pouvant être adapté (par ex., par ajustement fin) à un large éventail de tâches en aval."

de réduire les besoins globaux en stockage et la consommation énergétique en autorisant un modèle unique, bien que volumineux, à prendre en charge simultanément plusieurs tâches. De plus, le traitement en périphérie de réseau peut offrir une couche supplémentaire de sécurité en garantissant que les données sensibles sont traitées localement sur le dispositif plutôt que transmises vers des serveurs infonuagiques tiers.

Bien que les modèles de fondation constituent l'ossature permettant aux systèmes intelligents de percevoir le monde et de raisonner à partir des données, leur déploiement en périphérie de réseau demeure un goulot d'étranglement critique. Ces limitations comprennent notamment (sans s'y limiter):

- **Limitation #1: Capacité du modèle** — Les environnements en périphérie de réseau sont intrinsèquement contraints en termes de stockage et de puissance de calcul. En revanche, les modèles de fondation audio actuels [13] sont de très grande taille, comptant généralement entre 95 millions et 2 milliards de paramètres nombre à virgule flottante sur 32 bits (FP32). Cette disparité conduit les utilisateurs à explorer des techniques de compression de modèles; toutefois, si la compression rend le déploiement possible, elle s'accompagne souvent d'un coût. Nous observons que les modèles compressés, bien qu'ils conservent de hautes performances sur des bancs d'essai en conditions propres, deviennent significativement plus vulnérables aux menaces de sécurité et au bruit environnemental que leurs homologues de pleine capacité.
- **Limitation #2: Robustesse face aux dérives de distribution** — La capacité de généralisation des modèles d'apprentissage automatique repose généralement sur l'hypothèse d'indépendance et d'identité de distribution (independent and identically distributed, i.i.d.), selon laquelle les données d'entraînement et de test sont issues de la même distribution. Toutefois, cette hypothèse est fréquemment violée lorsque les modèles sont déployés "in-the-wild" et que des dérives de distribution surviennent, ce qui dégrade leurs performances. Par exemple, un système entraîné sur de la parole lue et propre (par ex., des livres audio) peut échouer lorsqu'il est confronté à de la parole spontanée dans des environnements bruyants. De manière cruciale, des preuves empiriques suggèrent que *les modèles compressés sont disproportionnellement sensibles* à ces dérives de distribution.
- **Limitation #3: Robustesse face aux exemples adverses** — Alors que les algorithmes d'apprentissage automatique cherchent à minimiser le risque empirique en identifiant l'hypothèse

la plus appropriée, les attaques adverses visent à maximiser la fonction de perte en concevant une petite perturbation, contrainte par la norme L_p de cette perturbation. Plus largement, une attaque adverse désigne un cadre dans lequel, au moment de l'inférence, un utilisateur malveillant cherche à provoquer un comportement du modèle différent de celui pour lequel il a été initialement conçu. Ces perturbations sont élaborées pour être faibles selon une métrique donnée, tout en étant suffisantes pour induire des prédictions erronées avec un haut degré de confiance. En outre, les attaques adverses peuvent être menées aussi bien dans le monde numérique que dans le monde physique, où l'entrée subit des transformations supplémentaires imposées par l'environnement ou par la chaîne d'acquisition des données [14].

Dans cette thèse, nous abordons le double défi consistant à compresser les modèles de fondation pour des environnements à ressources limitées tout en renforçant simultanément leur robustesse face aux données hors distribution. Nous étudions un large éventail de dérives de distribution, englobant à la fois des perturbations environnementales d'origine naturelle et des attaques adverses malveillantes. Plus précisément, notre objectif est de développer des méthodologies de compression qui préservent les performances initiales du modèle tout en démontrant une résilience significativement accrue dans des conditions de déploiement réelles en périphérie de réseau.

0.1.1 Domaines d'application et implications pratiques

Traitement de la parole

La parole constitue le mode fondamental de la communication humaine, ayant évolué au fil des millénaires comme un moyen efficace et expressif d'échange d'information [15, 16]. Elle représente par conséquent la modalité la plus intuitive pour améliorer l'interaction homme–machine. Contrairement au texte, qui est strictement linguistique, le signal de parole est riche en information: il véhicule non seulement le contenu sémantique (ce qui est dit), mais également des informations paralinguistiques abondantes, telles que l'état émotionnel du locuteur, l'intonation et l'identité [17]. De plus, des avancées récentes ont montré que la parole contient des marqueurs biologiques subtils, permettant une authentification biométrique non invasive ainsi que des applications de diagnostic en santé [18].

Compte tenu de ce large éventail d'applications, on observe un changement de paradigme croissant, porté par l'écosystème de l'Internet des objets (Internet of Things, IoT), vers le déploiement de

technologies de la parole directement sur des dispositifs en périphérie de réseau [19] (par ex., les assistants vocaux). Toutefois, le déploiement de ces modèles “in-the-wild” pose des défis considérables. Le développement de modèles à faible latence et à haute efficacité énergétique, capables de fonctionner de manière fiable sous des contraintes computationnelles strictes, est dès lors indispensable.

Surveillance bioacoustique

L’utilité du traitement audio économe en ressources s’étend au-delà de l’interaction humaine; elle est tout aussi essentielle pour la surveillance et la préservation du monde naturel. La biodiversité constitue le fondement de la stabilité écologique et de la santé planétaire, mais elle décline à un rythme sans précédent [20]. La perte accélérée des habitats, les changements climatiques et d’autres pressions d’origine anthropique ont conduit de nombreuses espèces vers la menace ou l’extinction, perturbant le fonctionnement des écosystèmes et mettant en péril les systèmes socioéconomiques qui en dépendent [21]. La surveillance, la préservation et la compréhension de la faune et de ses comportements ne relèvent donc pas uniquement de l’éthique écologique, mais constituent également un impératif mondial pour assurer la sécurité alimentaire et le bien-être humain à long terme [22].

Pour faire face à cette crise à grande échelle, la surveillance acoustique passive (Passive Acoustic Monitoring, PAM) s’est imposée comme une méthodologie puissante pour l’étude des sons produits par les animaux [23]. L’audio véhicule une information écologique riche [24]. De manière cruciale, tout comme dans le domaine de la parole, la faisabilité du PAM repose sur le paradigme de l’intelligence artificielle en périphérie de réseau (Edge AI). Avec l’essor de l’Internet des objets (IoT), les enregistreurs audio autonomes sont devenus abordables et déployables dans des environnements éloignés. Toutefois, ces capteurs génèrent des volumes massifs de données audio continues, rendant toute inspection manuelle irréaliste. Cela met en évidence un besoin urgent de méthodes automatiques de traitement du signal capables d’extraire des connaissances écologiques pertinentes directement sur le dispositif.

Cela engendre un défi technique commun: à l’instar du traitement de la parole, la surveillance bioacoustique requiert des modèles de fondation à la fois hautement efficaces et capables de généraliser sur une large plage dynamique de signaux. Étant donné que les animaux produisent une grande diversité de sons couvrant différentes fréquences et structures temporelles, ces modèles doivent être

suffisamment robustes pour gérer une forte variabilité, tout en restant suffisamment légers pour un déploiement à long terme sur des dispositifs à faible consommation énergétique.

0.1.2 Contributions de la thèse

Cette thèse s'attaque au défi du déploiement des modèles de fondation audio en périphérie de réseau en optimisant leur efficacité computationnelle tout en garantissant leur robustesse face aux conditions "in-the-wild". Les principales contributions de ce travail sont résumées comme suit:

- Nous proposons **RobustDistiller**, un cadre de distillation agnostique à la tâche, conçu pour compresser des modèles de fondation pour la parole tout en renforçant simultanément leur robustesse face à des perturbations environnementales non observées. Le modèle étudiant est entraîné à l'aide d'un objectif multi-tâches intégrant une nouvelle méthodologie, nommée *Feature Denoising Knowledge Distillation*, visant à imposer l'apprentissage de représentations invariantes au bruit. De plus, nous intégrons une tâche auxiliaire de reconstruction du signal afin de préserver l'information intrinsèque du signal et de favoriser la désintrication des caractéristiques. Des expériences menées sur 12 tâches de parole en aval montrent que RobustDistiller procure des gains significatifs dans des scénarios hors distribution, tout en améliorant également la généralisation dans le domaine.

- Nous introduisons **VIC-KD**, un cadre de distillation de connaissances robuste destiné à compresser les modèles tout en renforçant simultanément leur robustesse face aux attaques adverses bornées par la norme L_p. En nous concentrant sur des encodeurs de parole adaptés à la détection de mots-clés (*Keyword Spotting*), nous démontrons que l'imposition de priors géométriques sur les représentations latentes des modèles enseignant et étudiant conduit à des architectures significativement plus robustes. Les résultats expérimentaux obtenus sur le jeu de données *Google Speech Commands* indiquent que VIC-KD surpasse les méthodes de distillation robuste de l'état de l'art, en particulier ARD et RSLAD, en améliorant la précision robuste de 12 % et 8 %, respectivement.

- Nous étendons ensuite notre étude au domaine bioacoustique en présentant un banc d'essai exhaustif d'encodeurs audio généralistes et spécifiques au domaine pour la surveillance acoustique des ruches assistée par l'IoT, et ce même en présence de données annotées limitées. Quatorze

modèles sont évalués sur quatre tâches critiques: la détection de l'état de la ruche, l'évaluation de la vigueur de la colonie, l'identification du bourdonnement et la détection d'activité vocale de l'apiculteur. Les résultats rapportés mettent en évidence la forte capacité de généralisation des représentations existantes, ouvrant la voie à une surveillance avancée et évolutive des colonies d'abeilles.

- Enfin, nous introduisons **BioME**, un encodeur audio économe en ressources, spécifiquement conçu pour des applications bioacoustiques. BioME est entraîné par distillation couche par couche à partir d'un modèle enseignant à forte capacité, permettant un transfert de représentation efficace malgré une réduction de 75 % du nombre de paramètres. Afin d'améliorer la généralisation écologique, le modèle est préentraîné sur un corpus multi-domaines comprenant de la parole, des sons environnementaux et des vocalisations animales. Une contribution clé de ce travail réside dans l'injection d'un biais inductif inspiré du traitement du signal (DSP), intégrant des caractéristiques acoustiques sensibles à la modulation via un conditionnement FiLM, afin de faciliter la désintrication des caractéristiques au sein d'une architecture à faible capacité. Évalué sur 10 tâches bioacoustiques variées couvrant les oiseaux, les mammifères marins et les insectes, BioME égale ou dépasse les performances de modèles de référence plus volumineux, y compris son modèle enseignant, confirmant ainsi son adéquation pour des déploiements IoT à ressources limitées.

0.1.3 Organisation de la thèse

Cette thèse est structurée en deux grandes parties. Le Chapitre 2 présente le contexte nécessaire à la compréhension du reste de la thèse. La première partie est consacrée aux applications du traitement de la parole. Le Chapitre 3 introduit une méthodologie visant à traiter conjointement la compression des modèles et la robustesse aux dérives de distribution résultant de perturbations environnementales. Le Chapitre 4 prolonge cette discussion en proposant des approches pour concevoir des encodeurs de parole à la fois économes en ressources et robustes face aux attaques adverses. La seconde partie de la thèse est dédiée aux applications bioacoustiques. Le Chapitre 5 explore l'utilisation d'encodeurs de parole pour une tâche bioacoustique et introduit un banc d'essai pour la surveillance acoustique des ruches, dans le cadre duquel plusieurs encodeurs audio sont évalués. Le Chapitre 6 présente une méthodologie novatrice pour l'apprentissage d'un encodeur bioacoustique économe en ressources, exploitant des biais inductifs forts issus du traitement du signal (caractéristiques du spectre de

modulation) afin d'améliorer la généralisation dans des modèles à faible capacité. Enfin, le Chapitre 7 conclut la thèse en résumant les principales contributions et en discutant des limites ainsi que des pistes de recherche futures.

0.2 Résumé

0.2.1 Chapitre 2: Contexte

Réseaux neuronaux profonds

La première architecture importante pour ce travail est le réseau de neurones convolutif (Convolutional Neural Network, CNN), un modèle spécialisé conçu pour traiter des données présentant une topologie régulière connue, telles que des signaux audio unidimensionnels ou des spectrogrammes bidimensionnels [25]. Les biais inductifs associés aux CNN permettent au réseau de capturer efficacement des motifs locaux invariants par translation, tels que des frontières phonétiques ou des structures harmoniques, indépendamment de leur position dans la série temporelle.

La seconde architecture, et actuellement dominante, est le Transformer [26]. Contrairement aux opérations locales des CNN, les Transformers reposent sur le mécanisme d'auto-attention (*Self-Attention*) pour modéliser des dépendances globales. En considérant l'entrée comme une séquence de jetons, l'auto-attention permet au modèle de relier directement n'importe quelles positions du signal, indépendamment de leur distance temporelle. Cette capacité est cruciale pour capturer des informations contextuelles à longue portée, indispensables à la compréhension de la structure sémantique de haut niveau des flux de parole et d'audio.

Apprentissage auto-supervisé

Encodeurs de parole Dans nos travaux, trois modèles représentatifs sont principalement utilisés comme ossature pour les encodeurs de parole: Wav2Vec 2.0 [27], HuBERT [13] et WavLM [28]. Leur architecture se compose de deux blocs principaux: un encodeur convolutionnel de caractéristiques à plusieurs couches, qui projette l'audio brut vers des représentations latentes de la parole (généralement avec un pas temporel de 20 ms), et un réseau de contexte fondé sur des Transformers. Lors du

préentraînement, certaines représentations latentes sont masquées de manière aléatoire, à l'instar du *masked language modeling* (MLM).

Pour Wav2Vec 2.0, le modèle est optimisé à l'aide d'une tâche contrastive. HuBERT et WavLM, en revanche, s'appuient sur des tâches prétextes prédictives pour le préentraînement, en exploitant un regroupement hors ligne (*K-Means*) afin de générer des signaux de supervision stables. En outre, WavLM prolonge HuBERT afin d'améliorer la robustesse face à la variabilité environnementale.

Encodeurs audio généralistes Les encodeurs audio se répartissent généralement selon deux paradigmes architecturaux: les approches basées sur des trames (*frame-based*) et celles basées sur des patches (*patch-based*). Les modèles à base de trames, typiques du traitement de la parole, ingèrent des formes d'onde brutes afin de produire des séquences temporelles à haute résolution (par ex., 50 Hz). En revanche, les modèles audio à usage général adoptent souvent une approche par patches inspirée des Vision Transformers (ViT) [29], dans laquelle les spectrogrammes sont découpés en patches bidimensionnels aplatis.

BEATs [30] illustre cette seconde approche; ce modèle exploite un objectif de modélisation masquée pour prédire des étiquettes sémantiques discrètes à partir de patches de spectrogrammes masqués, démontrant ainsi une forte capacité de généralisation inter-domaines. À l'inverse, AVES [31] adapte l'architecture à base de trames (similaire à HuBERT) au domaine bioacoustique. Il traite des formes d'onde brutes au moyen d'un frontal convolutionnel afin d'extraire des caractéristiques à une fréquence de trames de 50 Hz, optimisées via le *Masked Language Modeling* (MLM) sur un large ensemble de données bioacoustiques.

Malgré leur succès, les encodeurs de parole et d'audio reposent sur des nombres de paramètres élevés, ce qui pose des défis pour leur déploiement sur des dispositifs en périphérie de réseau à ressources limitées. Bien que des modèles de petite taille existent, leurs performances demeurent inférieures à celles de leurs homologues de grande taille. Cette disparité met en évidence que la capacité du modèle reste un facteur déterminant pour l'apprentissage de représentations robustes. Il existe donc un besoin pressant d'architectures capables de combler cet écart, en conservant la puissance représentationnelle du préentraînement SSL à grande échelle tout en demeurant suffisamment légères pour des déploiements IoT en conditions réelles.

Évaluation de la qualité des représentations À mesure que les représentations universelles gagnent en popularité, des cadres d'évaluation standardisés deviennent indispensables pour permettre des comparaisons rigoureuses. Dans la communauté du traitement de la parole, le banc d'essai SUPERB [32] s'est imposé comme la référence de facto, en évaluant les représentations selon des dimensions liées au contenu, au locuteur, à la sémantique et aux aspects paralinguistiques. Ce cadre a récemment été étendu par SUPERB-SG [33], qui élargit l'évaluation aux tâches génératives de type signal-à-signal, telles que l'amélioration de la parole.

En parallèle, le banc d'essai BEANS [34] constitue la principale suite d'évaluation pour le domaine bioacoustique. Il regroupe 12 tâches diverses, allant de la classification des mammifères marins à la détection d'espèces menacées (par ex., le gibbon de Hainan), et est conçu pour évaluer la capacité de généralisation à travers de larges groupes taxonomiques, incluant notamment les oiseaux, les insectes et les chiroptères.

Compression de modèles

Quantification et élagage Le déploiement de réseaux neuronaux profonds sur des dispositifs en périphérie de réseau impose de fortes contraintes en termes de charge de calcul, de consommation énergétique et d'empreinte mémoire. Deux paradigmes de compression principaux dominent la littérature pour répondre à ces enjeux: la quantification et l'élagage. La quantification réduit la taille du modèle et la latence en inférence en diminuant la précision numérique des poids et des activations (par ex., en passant de nombres flottants sur 32 bits à des entiers sur 8 bits) [35, 36, 37]. À l'inverse, l'élagage vise à réduire la complexité architecturale en identifiant et en supprimant de manière systématique des paramètres ou des neurones redondants [38, 39, 40, 41].

Distillation de connaissances L'objectif principal de la distillation de connaissances (*Knowledge Distillation*, KD), initialement proposée par [42], est de transférer les connaissances d'un modèle enseignant à forte capacité, T_θ, vers un modèle étudiant compact, S_ϕ, permettant ainsi au réseau de plus petite taille d'approximer les capacités de généralisation du modèle plus volumineux. Alors que la KD classique se concentre sur les probabilités de sortie, les encodeurs de parole auto-supervisés, qui ne disposent généralement pas d'étiquettes de classes explicites, requièrent une distillation fondée sur

les caractéristiques. Les techniques visant à distiller ces représentations ont été largement étudiées dans la littérature récente [43, 44].

Dans le domaine de la parole, **DistilHuBERT** [45] illustre cette approche en atteignant environ 75 % de compression grâce à l'entraînement d'un modèle étudiant peu profond à deux couches, chargé d'approximer certaines couches internes spécifiques (par ex., les 4^{e}, 8^{e} et 12^{e}) du modèle enseignant au moyen de têtes de projection. À titre alternatif, **LightHuBERT** [46] combine la distillation avec un élagage structuré afin d'atteindre une efficacité de pointe; toutefois, son pipeline d'entraînement en deux étapes est coûteux sur le plan computationnel, ce qui le rend souvent prohibitif pour des ressources de type académique. Pour répondre à cette complexité, **DPHuBERT** [47] introduit un cadre simplifié de distillation et d'élagage conjoints, spécifiquement optimisé pour le banc d'essai SUPERB.

Adaptation de domaine

Malgré leurs performances impressionnantes sur des bancs d'essai standard, les encodeurs de parole présentent une sensibilité marquée aux conditions environnementales typiques des applications en périphérie de réseau. Par exemple, des travaux récents ont montré que lorsque les données de test sont altérées par des facteurs non observés lors de l'entraînement (par ex., la réverbération et le bruit), les dérives de distribution dégradent significativement les performances des modèles [48]. Afin d'atténuer ce problème, des techniques d'adaptation de domaine [49] ont été explorées.

À titre d'exemple, Robust HuBERT [50] propose d'exploiter l'entraînement adversarial de domaine (*Domain Adversarial Training*, DAT) [51] afin d'apprendre des caractéristiques invariantes au bruit, en utilisant le modèle HuBERT comme point de départ. Des améliorations notables de la robustesse dans des conditions de test bruitées ont ainsi été rapportées [50].

Attaques adverses

Définition et entraînement adversarial Alors que les algorithmes d'apprentissage automatique standards visent à minimiser la fonction de perte, les attaques adverses cherchent au contraire à la maximiser. Un adversaire construit une perturbation δ, contrainte à l'intérieur d'une boule de norme p dans l'espace métrique d'entrée, spécifiquement conçue pour induire une mauvaise

classification. Afin de se prémunir contre de telles menaces, la stratégie la plus répandue consiste à recourir à l'entraînement adversarial (*Adversarial Training*, AT) [52]. L'AT formalise le processus d'apprentissage comme un problème d'optimisation robuste de type min–max, dont l'objectif est de minimiser la perte dans le pire des cas générée par l'adversaire:

$$\min_{\theta} \hat{R}_{\text{adv}}(f_\theta, \mathcal{D}_{\text{train}}) = \min_{\theta} \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{train}}} \max_{\delta\in\Delta(x)} l(f_\theta(x+\delta), y), \tag{1}$$

où f_θ désigne le réseau neuronal paramétré par les poids θ, $l(\cdot,\cdot)$ représente la fonction de perte, et $\mathcal{D}_{\text{train}}$ correspond à l'ensemble de données d'entraînement contenant des paires entrée–étiquette (x, y).

En pratique, l'entraînement adversarial agit comme une technique rigoureuse d'augmentation de données, exposant continuellement le modèle à des exemples adverses générés dynamiquement au cours de l'optimisation. Bien que de nombreux travaux aient étudié la robustesse environnementale des encodeurs de parole, les recherches portant sur la robustesse adversariale demeurent relativement limitées. Wu et al. [53] ont notamment analysé la vulnérabilité d'encodeurs majeurs (par ex., Wav2Vec 2.0, HuBERT), en mettant en évidence leur sensibilité aux attaques itératives.

Distillation de connaissances robuste Bien que la distillation de connaissances soit efficace pour la compression de modèles, elle ne préserve pas intrinsèquement la robustesse adversariale du modèle enseignant, comme l'ont montré Goldblum et al. [54]. Pour pallier cette limitation, la *distillation adversarialement robuste* (*Adversarially Robust Distillation*, ARD) [54] a été proposée comme un cadre intégrant directement l'entraînement adversarial dans l'objectif de distillation. Le principe fondamental consiste à contraindre le modèle étudiant, lorsqu'il est exposé à des perturbations adverses, à aligner sa sortie sur celle du modèle enseignant correspondant à l'entrée propre.

De manière similaire, la *Robust Soft Label Adversarial Distillation* (RSLAD) [55] a émergé comme une stratégie alternative. RSLAD postule que le remplacement des étiquettes dures de vérité terrain par les étiquettes souples du modèle enseignant fournit un signal d'apprentissage plus riche lors de l'entraînement adversarial, conduisant empiriquement à une robustesse supérieure par rapport à ARD. Toutefois, comme nous le montrons au Chapitre 4, aucune de ces méthodes ne surpasse de manière systématique un modèle compact entraîné uniquement à l'aide de l'entraînement adversarial standard, dans le contexte des représentations de parole auto-supervisées.

0.2.2 Chapitre 3: Caractéristiques invariantes au bruit via distillation débruitante et apprentissage multi-tâches

Le Chapitre 3 aborde le double défi posé par le déploiement de représentations universelles de la parole: la dégradation significative des performances en présence de dérives de distribution environnementales (par ex., bruit et réverbération) et le coût computationnel prohibitif des modèles de fondation.

Pour répondre à ces enjeux, nous proposons **RobustDistiller**, un cadre agnostique à la tâche, conçu pour compresser conjointement des encodeurs de parole tout en renforçant leur résilience face aux conditions environnementales. Cette méthode repose sur un principe privilégiant la robustesse et peut être intégrée à des recettes de distillation existantes (par ex., DistilHuBERT). Le cadre introduit deux mécanismes visant à construire des modèles étudiants robustes:

1. **Distillation de connaissances par débruitage des caractéristiques:** Afin de contrer la dérive de domaine induite par le bruit environnemental, nous introduisons un mécanisme de contamination en ligne directement au sein de la chaîne de distillation. Inspirée du schéma d'entraînement de WavLM, cette méthode incite le modèle étudiant à reconstruire les représentations *propres* du modèle enseignant tout en traitant des entrées corrompues. Lors de l'entraînement, les énoncés de parole propres sont augmentés dynamiquement selon l'une des quatre stratégies stochastiques suivantes: (i) identité (propre); (ii) bruit additif (SNR $\sim \mathcal{U}[0, 20]$ dB); (iii) convolution avec une réponse impulsionnelle de salle aléatoire; ou (iv) combinaison simultanée de bruit et de réverbération. L'objectif de distillation, combinant une perte L_1 et une similarité cosinus, impose une contrainte de cohérence forte: quelle que soit la dégradation de l'entrée, l'espace latent du modèle étudiant doit s'aligner sur l'espace des caractéristiques propres du modèle enseignant. Notre hypothèse centrale est que cette incohérence entre l'entrée (bruitée) et la cible (propre) contraint le modèle à apprendre des caractéristiques invariantes au bruit.

2. **Reconstruction du signal par apprentissage multi-tâches:** Alors que la distillation se concentre sur l'espace latent, nous introduisons une tête auxiliaire d'amélioration du signal afin de régulariser le processus d'apprentissage. Motivée par des avancées récentes en apprentissage de représentations de la parole [56], cette tête vise à reconstruire la forme d'onde propre à

partir des représentations dégradées du modèle étudiant. Contrairement aux tâches classiques d'amélioration de la parole, qui optimisent principalement la qualité perceptuelle, notre objectif est ici de nature structurelle: garantir que l'encodeur en amont conserve une quantité suffisante d'information intrinsèque du signal, plutôt que de la perdre au cours de la compression. Cette approche d'apprentissage multi-tâches agit comme un régularisateur, empêchant le modèle de surajuster les cibles de distillation et favorisant la préservation de détails fins.

Configuration expérimentale

Jeux de données et augmentation Notre chaîne d'entraînement utilise le corpus **LibriSpeech** de 960 heures (parole propre) comme jeu de données principal. Afin de simuler des conditions "in-the-wild", nous appliquons une augmentation de données en ligne rigoureuse, en utilisant des jeux de données distincts pour l'entraînement et l'évaluation afin de garantir des scénarios strictement non observés.

- **Bruit d'entraînement:** Nous augmentons les données d'entraînement à l'aide de bruits non vocaux issus de MUSAN et d'UrbanSound8K (en excluant les catégories musique et enfants), ainsi que de réponses impulsionnelles de salle (Room Impulse Responses, RIR) réelles provenant d'OpenSLR28.
- **Bruit de test (non observé):** Pour l'évaluation, nous utilisons des sources entièrement distinctes: des scènes acoustiques issues du défi DCASE2020 (10 villes, 4 dispositifs) et des RIR simulées provenant d'OpenSLR26, couvrant des environnements allant de petites pièces à de grands auditoriums.

Préentraînement et modèles de référence Nous évaluons le cadre proposé en distillant quatre modèles enseignants distincts: Wav2Vec 2.0, HuBERT, WavLM et Robust HuBERT. Nous appliquons notre cadre RobustDistiller à ces ossatures, en ciblant les couches 4^{e}, 8^{e} et 12^{e}, conformément aux protocoles standard de la littérature [45]. Tous les modèles ont été entraînés sur une infrastructure unique équipée d'un NVIDIA A100.

Cadre d'évaluation Nous exploitons les bancs d'essai SUPERB et SUPERB-SG afin d'évaluer les performances sur 12 tâches en aval diverses, allant de la reconnaissance de contenu (ASR, PR) à la génération de signaux (amélioration de la parole). Pour tester rigoureusement la robustesse environnementale, nous définissons quatre protocoles d'évaluation: (i) propre (c), (ii) bruit seul (n) (SNR $\in [-5, 20]$ dB), (iii) réverbération seule (r), et (iv) bruit plus réverbération ($n + r$).

Les performances sont mesurées à l'aide de métriques spécifiques à chaque tâche (par ex., WER pour l'ASR, exactitude pour la classification, PESQ/STOI pour l'amélioration), ainsi que d'un "score SUPERB" agrégé afin de résumer les capacités globales. À noter que nous utilisons les hyperparamètres de fine-tuning par défaut de SUPERB afin de garantir une comparaison équitable, centrée exclusivement sur la qualité de l'encodeur en amont.

Discussion

Robustesse sur les tâches en aval Nos résultats montrent de manière systématique que les méthodes de compression standard (par ex., DistilHuBERT, FitHuBERT) subissent une dégradation catastrophique des performances dans des environnements bruités. En revanche, RobustDistiller atténue significativement cet écart. Par exemple, pour la tâche de détection de mots-clés (*Keyword Spotting*, KS) dans la condition exigeante $n + r$, notre méthode atteint une exactitude de 82,31 % (en distillant WavLM), surpassant la recette de distillation standard et dépassant même le modèle enseignant de pleine capacité. Cette tendance se vérifie sur l'ensemble des tâches liées au contenu, au locuteur et à la sémantique, pour lesquelles nos modèles étudiants de 23 M de paramètres égalent ou dépassent fréquemment la robustesse de modèles de référence comptant 95 M de paramètres.

Analyse mécanistique: pourquoi la méthode fonctionne Afin de comprendre l'origine de ces gains, nous avons analysé l'espace latent à l'aide de visualisations t-SNE. Alors que les modèles étudiants standards échouent à séparer les locuteurs sous un bruit de 0 dB SNR, RobustDistiller maintient des clusters distincts et désintriqués, confirmant notre hypothèse selon laquelle le débruitage des caractéristiques contraint le modèle à apprendre des représentations invariantes au bruit. Par ailleurs, les études d'ablation menées sur la tête d'amélioration multi-tâches montrent que la reconstruction d'un masque spectral binaire (à l'aide d'une perte L_1) fournit la régularisation la plus efficace. Cette tâche auxiliaire garantit que le modèle étudiant conserve l'information structurelle

du signal, évitant l'effondrement des caractéristiques fréquemment observé dans les schémas de compression purement discriminatifs.

Robustesse face aux variations environnementales Enfin, nous avons évalué les limites de notre modèle face à des facteurs environnementaux plus fins. L'analyse des régimes à faible SNR révèle que RobustDistiller demeure stable face à une grande diversité de types de bruit (intérieur, extérieur, transport), tandis que les modèles de référence échouent de manière marquée en présence de bruit intérieur. De même, lors d'expériences menées avec des tailles de pièces variables, notre modèle conserve une exactitude élevée, y compris dans de grands auditoriums (RT60 élevé), où les modèles standards voient leurs performances se dégrader significativement.

En résumé, le Chapitre 3 détaille les fondements techniques de RobustDistiller, une méthodologie visant à améliorer la robustesse environnementale du processus de distillation pour l'apprentissage de représentations de la parole. Les résultats expérimentaux montrent que la méthode proposée surpasse fréquemment les modèles enseignants originaux tout en utilisant environ quatre fois moins de paramètres. Globalement, l'application de RobustDistiller au schéma DPWavLM conduit aux meilleures performances sur l'ensemble des tâches considérées, suggérant ainsi que cette approche est particulièrement adaptée aux applications de parole embarquées en conditions "in-the-wild".

0.2.3 Chapitre 4: Caractéristiques robustes aux attaques adverses via l'analyse de corrélation canonique

Les systèmes de détection de mots-clés (*Keyword Spotting*, KWS) constituent généralement la première couche des assistants virtuels, où ces modèles cherchent à identifier des mots prédéfinis dans un énoncé afin d'activer le dispositif (par ex., "Hey Siri"). Le traitement de la parole embarqué, tel que celui employé par Google Now ou Alexa, devient omniprésent dans notre quotidien; toutefois, le déploiement de ces systèmes se heurte à un goulot d'étranglement critique en matière de sécurité: leur vulnérabilité aux attaques adverses. Des travaux récents indiquent que les systèmes KWS reposant sur des ossatures SSL sont particulièrement sensibles à des perturbations malveillantes lors de l'inférence [57].

Dans le Chapitre 4, nous proposons **VIC-KD**, un nouveau cadre de distillation qui compresse les encodeurs de parole tout en renforçant de manière significative leur robustesse adversariale face

aux attaques numériques. En particulier, nous montrons que l'intégration d'entrées multi-vues et l'imposition de contraintes géométriques sur l'espace latent du modèle étudiant sont essentielles pour combler l'écart de robustesse. Nous nous plaçons dans un régime de compression extrême, en ciblant des modèles étudiants comptant moins de 96 k paramètres et 3 millions d'opérations de type MAC. En distillant à partir de versions ajustées finement de *Wav2Vec 2.0* et de WavLM, nos résultats montrent que VIC-KD atteint une robustesse supérieure à celle de défenses classiques telles que TRADES, et surpasse les méthodes de distillation robuste de l'état de l'art (ARD et RSLAD).

Pour VIC-KD, plutôt que de s'appuyer uniquement sur le modèle enseignant pour guider le comportement des logits du modèle étudiant, nous induisons des propriétés géométriques de l'espace latent du modèle étudiant au moyen d'une régularisation variance–invariance–covariance (Variance-Invariance-Covariance Regularization, VICReg) [58] et de l'utilisation d'entrées multi-vues pour chaque modèle. Dans un premier temps, nous échantillonnons un énoncé du jeu de données d'entraînement ainsi que deux transformations aléatoires $\{t, t' \sim \mathcal{T} \mid t \neq t'\}$. Le cadre proposé opère ensuite selon deux branches parallèles:

- **Branche enseignant:** extrait des représentations propres à partir d'une vue augmentée de l'entrée.
- **Branche étudiant:** reçoit l'autre vue de l'énoncé générée par t', à laquelle est ajoutée une perturbation adverse produite à l'aide d'une descente de gradient projetée en 10 étapes (*Projected Gradient Descent*, PGD) [52].

L'innovation centrale réside dans la fonction de perte, qui intègre la régularisation variance–invariance–covariance (VICReg) [58] au processus de distillation. Cet objectif incite le modèle étudiant à apprendre des représentations qui soient non seulement invariantes au bruit adversarial, mais également à forte variance (informatives) et décorrélées (désintriquées). En imposant ces priors géométriques, VIC-KD empêche "l'effondrement des caractéristiques" (*feature collapse*) fréquemment observé lors de l'entraînement robuste.

Configuration expérimentale

Jeux de données et architecture Nous standardisons notre évaluation sur le jeu de données *Google Speech Commands* v0.02 (GSC) [59], en utilisant la partition officielle de SpeechBrain [60] (répartition 80/10/10). Afin de tester rigoureusement notre objectif de compression, nous distillons à partir de modèles enseignants auto-supervisés (*Wav2Vec 2.0* et *WavLM*) vers deux architectures étudiantes ultra-légères: TC-ResNet-8 [61] et un XVector de petite taille [62]. Les deux modèles étudiants sont optimisés avec un nombre réduit de couches et de tailles de noyaux, de manière à respecter strictement les contraintes de ressources propres aux dispositifs en périphérie de réseau à faible consommation énergétique.

Modèle de menace et modèles de référence Nous définissons des modèles de menace adverses bornés par une contrainte de norme ℓ_∞ de $\epsilon = 1{,}5 \times 10^{-3}$. Dans ce cadre, nous évaluons la robustesse face à un ensemble exhaustif d'attaques, allant des méthodes à un seul pas de gradient (FGSM) aux attaques itératives en boîte blanche (PGD-30, AutoAttack), ainsi qu'aux attaques en boîte noire fondées sur la prise de décision (FAKEBOB).

Afin de valider l'efficacité de VIC-KD, nous le comparons à la distillation de connaissances standard (Knowledge Distillation, KD), ainsi qu'aux cadres de distillation robuste de l'état de l'art, en particulier ARD et RSLAD. Le pipeline d'entraînement proposé enrichit ces lignes de base en intégrant une stratégie d'augmentation multi-vues riche, comprenant du bruit additif, de la réverbération, des perturbations de vitesse et la suppression de segments de forme d'onde (*wave chunk dropping*), ainsi qu'un adversaire PGD à 10 étapes appliqué au cours de l'optimisation afin de favoriser un apprentissage diversifié et robuste.

Discussion

Constat #1: Supériorité de la régularisation géométrique Notre analyse principale met en évidence l'échec critique de la distillation de connaissances standard (Knowledge Distillation, KD) en contexte adversarial: bien qu'elle atteigne une exactitude élevée en conditions propres, ses performances s'effondrent jusqu'à un niveau proche du hasard sous attaque, démontrant que la robustesse n'est pas naturellement héritée du modèle enseignant. Les méthodes robustes existantes,

telles que ARD et RSLAD, apportent des améliorations, mais peinent souvent à surpasser une ligne de base supervisée entraînée avec entraînement adversarial (AT). À l'inverse, VIC-KD surpasse systématiquement l'ensemble des modèles de référence. Pour le modèle étudiant XVector, VIC-KD dépasse l'exactitude robuste de la ligne de base TRADES de 7,2 % et surpasse ARD et RSLAD de 8,0 % et 6,6 %, respectivement, sous AutoAttack. Fait notable, nous observons que la distillation à partir d'un modèle enseignant *standard* est suffisante: transformer préalablement l'enseignant en un "enseignant robuste" n'apporte que des gains négligeables pour le modèle étudiant, suggérant que la distillation robuste doit avant tout se concentrer sur le processus lui-même.

Constat #2: Rôle des entrées multi-vues L'introduction de vues diversifiées (par ex., bruit, réverbération, perturbation de vitesse) améliore la robustesse pour l'ensemble des méthodes de distillation. De manière cruciale, des approches telles que ARD et RSLAD nécessitent l'utilisation des entrées multi-vues simplement pour atteindre le niveau de la ligne de base robuste. VIC-KD, en revanche, se distingue: il dépasse la ligne de base même sans entrées multi-vues, tout en atteignant des performances de l'état de l'art lorsque celles-ci sont utilisées. Ainsi, contraindre le modèle étudiant à maximiser l'information mutuelle entre les vues propres et perturbées permet de désintriquer le contenu sémantique de la parole du bruit adversarial.

Constat #3: Scalabilité et résilience Enfin, nous avons évalué la robustesse du cadre proposé dans des conditions de difficulté croissante.

- **Intensité des attaques:** Les études d'ablation montrent que, lorsque l'intensité des attaques augmente (valeurs de ϵ plus élevées ou nombre accru d'itérations PGD), VIC-KD conserve un avantage de performance marqué par rapport aux méthodes concurrentes, avec une dégradation nettement plus progressive que ARD ou RSLAD.
- **Complexité des tâches:** Lors du passage de la tâche GSC à 12 classes à celle à 35 classes, VIC-KD présente l'écart le plus faible entre l'exactitude en conditions propres et l'exactitude robuste, confirmant son adéquation pour des scénarios complexes et proches des conditions réelles.

0.2.4 Chapitre 5: Exploiter les modèles de fondation pour la surveillance acoustique des ruches

Nous orientons ensuite notre attention vers les applications bioacoustiques au Chapitre 5. Nous commençons par proposer un banc d'essai dédié à la surveillance acoustique des ruches. Les abeilles domestiques (*Apis mellifera*) jouent un rôle crucial dans le maintien de l'équilibre écologique et de la biodiversité en tant que pollinisateurs efficaces, contribuant à la reproduction de nombreuses espèces végétales. Elles présentent également une importance économique majeure, par la production de miel et de cire d'abeille, et influencent environ un tiers de la production alimentaire mondiale [63]. En résumé, l'objectif de ce chapitre est d'aider les chercheurs et les praticiens en apiculture à identifier rapidement un point de départ pertinent pour l'analyse des ruches fondée sur des méthodes d'apprentissage automatique.

Dans ce chapitre, nous proposons une évaluation standardisée des performances des représentations audio pour des tâches de surveillance acoustique des ruches. Notre pipeline d'évaluation (banc d'essai) repose sur trois composants principaux: (1) l'*UpstreamExpert*, un module englobant un modèle audio auto-supervisé chargé de l'extraction de caractéristiques; (2) le *Featurizer*, qui agrège les représentations issues de plusieurs couches de l'UpstreamExpert; et (3) le *DownstreamBrain*, qui exploite ces représentations pour des tâches spécifiques.

Tâches en aval Le module en aval regroupe un ensemble de composants chargés de prendre les représentations extraites et de les appliquer à une tâche spécifique. Étant donné que les modèles en amont sont entraînés de manière auto-supervisée, l'évaluation des représentations sur plusieurs tâches constitue un moyen de vérifier si ces modèles apprennent à extraire des caractéristiques "universelles" et pertinentes directement à partir des enregistrements. Dans ce cadre, quatre tâches en aval ont été sélectionnées pour l'évaluation:

- **États de la ruche (BSTS)**: Étant donné un enregistrement provenant d'une ruche, cette tâche vise à déterminer si la reine est présente dans l'échantillon. L'identification de l'absence de la reine est fondamentale pour la gestion apicole [64]. La reine joue un rôle essentiel dans la stabilité de la colonie, allant de la reproduction, nécessaire à l'augmentation du nombre d'ouvrières, à la production de phéromones assurant la cohésion sociale.

- **Vigueur de la ruche (BSTR)**: Cette tâche vise à prédire la vigueur d'une ruche. À l'instar d'autres travaux [65, 66], nous utilisons le nombre de cadres occupés par les abeilles comme indicateur de la vigueur de la colonie. Traditionnellement, des apiculteurs expérimentés inspectent visuellement chaque cadre afin d'évaluer la taille de la colonie et son état de santé global.
- **Identification du bourdonnement (BID)**: Dans cette tâche, les enregistrements audio sont classés en trois catégories: bourdonnement des abeilles, stridulation des grillons ou bruits naturels ambiants. La tâche BID peut servir d'étape de prétraitement permettant de filtrer les enregistrements et d'éliminer les artefacts, afin de déterminer leur pertinence pour les analyses ultérieures de la chaîne de traitement.
- **Détection d'activité vocale (VAD)**: La dernière tâche consiste à déterminer la présence ou non de parole humaine dans un enregistrement provenant d'une ruche.

Dans l'ensemble, les jeux de données et les tâches présentent une grande variabilité des conditions d'enregistrement. Par exemple, le jeu de données BSTS comprend des enregistrements contenant de la parole en arrière-plan, des bruits de circulation et des sons d'oiseaux. Le positionnement du microphone constitue un autre facteur de variation d'un jeu de données à l'autre. Notre travail se concentre sur l'évaluation d'encodeurs audio auto-supervisés à usage général, conçus pour apprendre des représentations idéalement robustes aux variations des conditions acoustiques, y compris celles liées au placement du microphone.

Résultats expérimentaux et discussion

Notre analyse nous a permis de dégager plusieurs enseignements clés concernant le déploiement de modèles auto-supervisés pour la surveillance écologique. Tout d'abord, nous observons que, si les modèles entraînés sur des corpus audio diversifiés fournissent une initialisation solide, une adaptation spécifique au domaine (par ex., un préentraînement continu à l'aide de données de bourdonnement d'abeilles) peut améliorer de manière significative les performances sur les tâches en aval.

Ensuite, nous mettons en évidence un compromis crucial : bien que la majorité des modèles de fondation excellent dans des scénarios en distribution, la généralisation hors distribution demeure

un défi majeur, les caractéristiques conçues manuellement constituant parfois des lignes de base compétitives pour des tâches complexes de classification d'états.

Du point de vue de l'efficacité en ressources, nos résultats soulignent la pertinence d'architectures compactes destinées au calcul en périphérie de réseau. En particulier, des modèles de petite taille tels que SS-AST (Small) atteignent des niveaux de précision compétitifs malgré une empreinte réduite, confirmant leur adéquation pour des dispositifs IoT à faible consommation énergétique. Ces résultats ouvrent la voie à la phase finale de la thèse, en motivant l'exploration de techniques de distillation avancées combinant efficacité et connaissances expertes afin de mieux concevoir des modèles étudiants performants, sujet qui sera abordé au Chapitre 6.

0.2.5 Chapitre 6 : BioME : un encodeur neuronal bioacoustique sensible aux modulations

Dans le chapitre précédent, nous avons discuté du rôle crucial des abeilles domestiques dans la stabilité écologique et la sécurité alimentaire, tout en évaluant comment des modèles de fondation acoustiques largement utilisés peuvent être exploités pour diverses tâches de surveillance des ruches. Toutefois, cette problématique dépasse le seul cas des abeilles : la biodiversité dans son ensemble connaît un déclin sans précédent en raison des changements climatiques et des activités anthropiques [21]. Pour faire face à cette situation, la surveillance acoustique passive (Passive Acoustic Monitoring, PAM) s'est imposée comme une méthodologie puissante en bioacoustique, permettant l'étude des sons produits par les animaux [23] de manière non invasive, sans perturber leur comportement ni la dynamique des écosystèmes.

Dans ce contexte, le Chapitre 6 introduit **BioME** (*Bioacoustic Modulation-aware Encoder*), une famille d'architectures économes en ressources, conçues spécifiquement pour permettre la surveillance acoustique passive sur des dispositifs en périphérie de réseau. BioME est entraîné par distillation de connaissances à partir d'un modèle enseignant BEATs à forte capacité vers un modèle étudiant compact. Afin d'améliorer l'efficacité, nous intégrons des optimisations architecturales issues de la littérature sur les grands modèles de langage (LLM), telles que l'attention à requêtes groupées (*Grouped Query Attention*, GQA) et les encodages positionnels rotatifs (*Rotary Positional Embeddings*, RoPE).

En outre, nous introduisons un nouveau biais inductif dans le processus de distillation : la fusion de caractéristiques issues du spectre de modulation en tant qu'information de canal auxiliaire à chaque couche du Transformer. Cette conception exploite le fait que les signaux bioacoustiques et le bruit environnemental, bien qu'ils se recouvrent souvent dans les domaines temporel et fréquentiel, sont généralement séparables dans le domaine des modulations en fonction de leurs taux de variation. Elle permet ainsi de tirer parti de la nature périodique des vocalisations animales.

Configuration expérimentale

Jeux de données Nous entraînons BioME en nous appuyant sur des jeux de données standards de la littérature, de manière similaire à AVES [31], en combinant AudioSet, FSD50K et VGGSound. En complément, le jeu de données iNatSounds [67] est également utilisé afin d'accroître la représentation des données bioacoustiques, couvrant plus de 5 500 espèces (principalement des vocalisations d'oiseaux), et visant à spécialiser l'encodeur pour des tâches de surveillance écologique. Toutes les entrées audio sont normalisées à une fréquence d'échantillonnage de 16 kHz.

Architecture et entraînement Afin d'évaluer la scalabilité sous différentes contraintes de ressources, nous avons développé trois variantes du modèle : BioME Edge (6 M), Small (26 M) et Base (76 M). L'ensemble des modèles a été entraîné pendant 100 000 itérations sur un seul GPU NVIDIA H100, en utilisant l'optimiseur AdamW avec une stratégie de taux d'apprentissage à décroissance cosinus (atteignant un maximum de 10^{-4}).

L'efficacité de ces architectures est particulièrement manifeste au moment de l'inférence : le traitement de 5 minutes d'audio continu ne nécessite que 1,7 Go, 3,6 Go et 4,3 Go de mémoire maximale (FP32) pour les variantes Edge, Small et Base, respectivement, confirmant ainsi leur faisabilité pour des déploiements sur des matériels variés.

Discussion

Nous avons validé BioME sur le banc d'essai BEANS ainsi que sur notre benchmark spécifique de surveillance acoustique des ruches, montrant qu'il permet de rompre systématiquement le compromis traditionnel entre la taille du modèle et la précision. Comparé à des lignes de base de capacité

similaire, BioME affiche des gains substantiels ; en particulier, la variante BioME Edge (6 M) surpasse BYOL-A (5 M) de 7,5 %, tandis que BioME Small (26 M) dépasse SS-AST (23 M) de 17,6 %. Fait remarquable, nos modèles distillés surpassent des modèles de fondation nettement plus volumineux (par ex., AVES et BEATs) sur des tâches hors distribution exigeantes, telles que la classification de l'état des ruches, où les distributions d'entraînement et de test proviennent de ruches différentes.

Enfin, nous avons analysé plus en profondeur notre hypothèse selon laquelle un conditionnement sensible aux modulations favorise un meilleur apprentissage des représentations. Les visualisations de cartes de saillance révèlent que BioME porte une attention plus fine aux structures harmoniques et aux formants que les modèles de référence ablatés. De plus, l'analyse de l'espace latent sur un jeu de données de colibris en zéro-shot montre que BioME produit une désintrication des caractéristiques significativement supérieure à celle d'encodeurs concurrents. Ces résultats confirment que l'intégration de biais inductifs issus du traitement du signal, à partir de caractéristiques de modulation, constitue un levier clé pour le déploiement de systèmes bioacoustiques robustes et performants en conditions réelles.

Chapter 1

Introduction

Recent advances in the deep (machine) learning field have profoundly impacted the speech- and audio-processing communities. As such, it has resulted in cutting-edge systems for tasks like automatic speech recognition [2, 3], speaker verification [4], biodiversity monitoring [5, 6], and many others. Usually, these systems rely on large neural networks that transform complex input data into more manageable representations [7]. While supervised learning has yielded specialized models that excel at their target tasks, this approach remains inherently constrained by the scarcity of human annotations. This bottleneck is particularly acute in specialized domains. For example, fields such as bioacoustics might require several specialists' hours to categorize a relatively small amount of data (e.g., identifying birds from a year's worth of audio recordings).

To address this limitation, recent literature focuses on developing generalist models capable of performing multiple tasks and acquiring new skills without relying on massive amounts of labeled data. In this context, instead of merely learning a rigid mapping from input to output, the objective is to create generalized predictive models that characterize objects and signals by observing the world, thereby learning to abstract high-level concepts (i.e., features or representations).

Learning to create meaningful and disentangled representations from high-dimensional signals, without explicit supervision, is crucial to this research. While fully characterizing *meaningful* representations is an ill-defined problem, some properties are desirable. Firstly, representations should be independent of redundant information in the data to capture different variations effectively [8]. Secondly, representations should be disentangled, meaning they separate different factors in the data,

making them easier to apply to multiple downstream tasks [9]. For example, for acoustic signals, these learned features capture essential aspects of speech and/or audio, such as phonemes, dynamic representation of harmonics, and/or spectral transitions [10].

Self-supervised learning (SSL) has emerged as the dominant paradigm for learning representations from vast amounts of unlabeled data. The core principle of SSL is to exploit the underlying structure of the data to generate supervisory signals. To train a foundation model[1] capable of encoding generalizable concepts, we typically rely on predicting hidden parts of the signal from observed contexts (e.g., masked language modeling [12]). This objective serves as a *pretext task*, where the primary goal is not the reconstruction itself, but the extraction of robust explanatory factors. In addition, to assess the quality of these learned representations, linear probing is the most widely adopted methodology. When combined with established benchmarks, probing provides a rigorous and consistent framework for verifying performance across diverse downstream applications.

Lastly, foundation models are a promising alternative for real-world deployment. By serving as a unified backbone for diverse applications, they obviate the need to train and maintain distinct high-capacity models for every specific task, as shared representations are reused across solutions. This reuse reduces overall storage requirements and power consumption by enabling a single, yet large, model to support multiple tasks simultaneously. Moreover, when foundation models are deployed "at the edge," this can provide an additional layer of security by ensuring sensitive data is handled locally on the device rather than transmitted to third-party cloud servers.

Having this said, while foundation models are the backbone that enable intelligent systems to perceive and reason about the world through data, deploying these models to the edge remains a critical bottleneck. These limitations include (but are not limited to):

- **Limitation #1: Model Capacity -** Edge environments are inherently constrained in storage and processing power. In contrast, current audio foundation models [13] are massive, typically ranging from 95 million to 2 billion parameters in 32-bit floating-point (FP32) precision. This disparity leads users to explore model compression. However, while compression enables deployment, it often comes at a cost. We observe that compressed models, despite maintaining

[1]Throughout this thesis, we will use the terms 'SSL model', 'Speech and Audio Encoders', and 'Foundation model' interchangeably, following the definition of [11]: "A foundation model is any model that is trained on broad data (generally using self-supervision at scale) that can be adapted (e.g., fine-tuned) to a wide range of downstream tasks."

high performance on clean benchmarks, become significantly more vulnerable to security threats and environmental noise compared to their full-sized counterparts.

- **Limitation #2: Robustness against distribution shifts -** The generalization of machine learning models typically relies on the independent and identically distributed (i.i.d.) assumption, under which training and test data are drawn from the same distribution. However, this assumption is often violated when models are deployed "in-the-wild" and distribution shifts occur, hampering model performance. For instance, a system trained on clean and read speech (e.g., audiobooks) may fail when encountering spontaneous speech in noisy environments. Crucially, empirical evidence suggests that *compressed models are disproportionately sensitive* to these distribution shifts.

- **Limitation #3: Robustness against adversarial examples -** While machine learning algorithms aim to minimize the empirical risk by finding the most suitable hypothesis, adversarial attacks maximize the loss function by designing a small perturbation constrained to the L_p-norm of this perturbation. More broadly, an adversarial attack is a general concept in which, at test time, a malicious user seeks to make a model behave in a manner for which it was not originally intended. These perturbations are designed to be small according to a chosen metric, yet sufficient to cause confident mispredictions. Furthermore, adversarial attacks can be carried out in the digital or physical world, where the input undergoes additional transformations imposed by the environment or the data-acquisition pipeline [14].

In this thesis, we address the dual challenge of compressing foundation models for resource-constrained environments while simultaneously enhancing their robustness to out-of-distribution data. We investigate a broad spectrum of distribution shifts, encompassing both naturally occurring environmental perturbations and malicious adversarial attacks. Specifically, our objective is to develop compression methodologies that preserve the model's original performance while demonstrating significantly improved resilience under real-world edge-deployment conditions.

1.1 Application domains and practical implications

1.1.1 Speech processing

Speech constitutes the fundamental mode of human communication, evolving over millennia as an efficient and expressive medium for information exchange [15, 16]. Consequently, it is the most intuitive modality for enhancing human-computer interaction. Unlike text, which is strictly linguistic, the speech signal is information-dense: it carries not only the semantic content (what is said) but also rich paralinguistic information, such as the speaker's emotional states, intonation, identity, and other demographic details, such as age and biological sex [17]. Moreover, recent advances have demonstrated that speech contains subtle biological markers, enabling non-invasive biometric authentication and health diagnostics [18].

Given this broad spectrum of applications, there is a growing paradigm shift, driven by the Internet of Things (IoT) community, towards deploying speech technologies directly onto edge devices [19] (e.g., smart speakers). However, deploying these models "in-the-wild" presents significant challenges. The development of low-latency, energy-efficient models that operate reliably within strict computational budgets is a must.

1.1.2 Bioacoustic monitoring

The utility of resource-efficient audio processing extends beyond human interaction; it is equally critical for monitoring and preserving the natural world. Biodiversity is the foundation of ecological stability and planetary health, yet it is declining at an unprecedented rate [20]. Accelerated habitat loss, climate change, and other anthropogenically driven pressures have pushed numerous species toward endangerment or extinction, disrupting ecosystem functioning and threatening the socio-economic systems that depend on it [21]. Monitoring, preserving, and understanding wildlife and its behavior are therefore not only a matter of ecological ethics but also a global imperative for maintaining food security and long-term human well-being [22].

To address this crisis at scale, passive acoustic monitoring (PAM) has emerged as a powerful methodology for studying animal-produced sounds [23]. Audio carries rich ecological information [24]. Crucially, just as in the speech domain, the feasibility of PAM relies on the Edge AI paradigm. With

the advent of the IoT, autonomous audio recorders have become affordable and deployable in remote environments. However, these sensors generate massive volumes of continuous audio data, rendering manual inspection infeasible. This highlights an urgent need for automatic signal processing methods capable of extracting meaningful ecological knowledge directly on the device.

This creates a shared technical challenge: much like speech processing, bioacoustic monitoring requires foundation models that are highly efficient, yet capable of generalizing to a vast dynamic range of signals. Since animals produce a diverse array of sounds spanning different frequencies and temporal structures, these models must be robust enough to handle high variability while remaining lightweight enough for long-term deployment on low-power-consumption devices.

1.2 Thesis contributions

This thesis addresses the challenge of deploying audio foundation models to the edge by optimizing their computational efficiency while ensuring robustness against "in-the-wild" conditions. The key contributions of this work are summarized as follows:

- We propose RobustDistiller, a task-agnostic distillation framework designed to compress speech foundation models while simultaneously enhancing their robustness against unseen environmental perturbations. The Student model is trained via a multi-task objective that incorporates a new methodology, named Feature Denoising Knowledge Distillation, to enforce the learning of noise-invariant representations. Furthermore, we integrate a signal reconstruction auxiliary task to preserve intrinsic signal information and promote feature disentanglement. Experiments across 12 different speech tasks demonstrate that RobustDistiller yields significant gains in out-of-distribution scenarios while also improving in-domain generalization.
- We introduce VIC-KD, a robust knowledge distillation framework designed to compress models while simultaneously enhancing their robustness against L_p-norm adversarial attacks. Focusing on speech encoders adapted for keyword spotting, we demonstrate that imposing geometric priors on the latent representations of both Teacher and Student models yields significantly more robust architectures. Experimental results on the Google Speech Commands dataset indicate that VIC-KD outperforms current state-of-the-art robust distillation methods, specifically ARD and RSLAD, improving robust accuracy by 12% and 8%, respectively.

- We extend our investigation into the bioacoustic domain, presenting a comprehensive benchmark of general-purpose and domain-specific audio encoders for IoT-enabled acoustic beehive monitoring, even with limited labeled data. Fourteen models are evaluated across four critical tasks: beehive state detection, beehive strength assessment, buzzing identification, and beekeeper voice activity detection. Reported results demonstrate the strong generalizability of existing representations, paving the way for advanced, scalable honeybee colony monitoring.

- Lastly, we introduce BioME, a resource-efficient audio encoder tailored for bioacoustic applications. BioME is trained via layer-to-layer distillation from a high-capacity Teacher, achieving strong representational transfer even with a 75% reduction in parameters. To enhance ecological generalization, the model is pretrained on a multi-domain corpus comprising speech, environmental sounds, and animal vocalizations. Here, a key contribution is the injection of a DSP-inspired inductive bias, integrating modulation-aware acoustic features via FiLM conditioning to facilitate feature disentanglement within a low-capacity architecture. Evaluated across 10 diverse bioacoustic tasks spanning birds, marine mammals, and insects, BioME matches or exceeds the performance of larger baselines, including its Teacher, confirming its suitability for resource-constrained IoT deployments.

1.3 List of publications

Publications included in the thesis

(**J1**) **Heitor R. Guimarães**, Abhishek Tiwari, Mahsa Abdollahi, Anderson Avila, and Tiago H. Falk. "BioME: A Resource-Efficient Bioacoustic Foundational Model for IoT Applications." Submitted to the IEEE Journal of Selected Areas in Sensors (JSAS) (2026).

(**J2**) **Heitor R. Guimarães**, Mahsa Abdollahi, Yi Zhu, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "Benchmarking Self-Supervised Audio Representations for IoT-enabled Acoustic Beehive Monitoring." IEEE Internet of Things Journal (IEEE IoT-J) (2025).

(**C1**) **Heitor R. Guimarães**, Arthur Pimentel, Anderson Avila, and Tiago Falk. "VIC-KD: Variance-Invariance-Covariance Knowledge Distillation to Make Keyword Spotting More Robust Against

Adversarial Attacks." IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) (2024).

(**C2**) **Heitor R. Guimarães***, Mahsa Abdollahi*, Yi Zhu, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "Adapting Self-Supervised Features for Background Speech Detection in Beehive Audio Recordings." IEEE International Workshop on Metrology for Agriculture and Forestry (2023).

Best Paper Presented by a Young Researcher Award

(**C3**) **Heitor R. Guimarães**, Yi Zhu, Orson Mengara, Anderson R. Avila, and Tiago H. Falk. "Assessing the Vulnerability of Self-Supervised Speech Representations for Keyword Spotting Under White-Box Adversarial Attacks." IEEE International Conference on Systems, Man, and Cybernetics (SMC) (2023).

(**C4**) **Heitor R. Guimarães**, Arthur Pimentel, Anderson Avila, Mehdi Rezagholizadeh, Boxing Chen, and Tiago Falk. "RobustDistiller: Compressing Universal Speech Representations for Enhanced Environment Robustness." IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) (2023).

(**C5**) **Heitor R. Guimarães**, Arthur Pimentel, Anderson Avila, Mehdi Rezagholizadeh, and Tiago Falk. "Improving the Robustness of DistilHuBERT to Unseen Noisy Conditions via Data Augmentation, Curriculum Learning, and Multi-Task Enhancement". Efficient Natural Language and Speech Processing (ENLSP-II) Workshop held at Advances in Neural Information Processing Systems (NeurIPS) (2022).

(**C6**) **Heitor R. Guimarães**, Arthur Pimentel, Anderson Avila, Mehdi Rezagholizadeh, and Tiago Falk. "An Exploration into the Performance of Unsupervised Cross-Task Speech Representations for *In the Wild* Edge Applications". Edge Intelligence Workshop (2022).

Publications from other collaborative projects

(**J3**) Mahsa Abdollahi, Yi Zhu, **Heitor R. Guimarães**, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "Audio Modulation Spectral Features for Improved Honeybee Colony Population Prediction." IEEE Sensors Journal (2025).

(**J4**) Mahsa Abdollahi, Yi Zhu, **Heitor R. Guimarães**, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "UrBAN: Urban Beehive Acoustics and PheNotyping Dataset." Nature Scientific Data (2025).

(**J5**) Yi Zhu, Mahsa Abdollahi, **Heitor R. Guimarães**, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "MSPB: a longitudinal multi-sensor dataset with phenotypic trait measurements from honey bees." Nature Scientific Data (2024).

(**J6**) Wesley Beccaro, Miguel Arjona Ramírez, William Liaw, and **Heitor R. Guimarães**. "Analysis of Oral Exams with Speaker Diarization and Speech Emotion Recognition: A Case Study." IEEE Transactions on Education (2023).

(**J7**) Arthur Pimentel, **Heitor R. Guimarães**, Anderson Avila, and Tiago Falk. "Environment-Aware Knowledge Distillation for Improved Resource-Constrained Edge Speech Recognition." Applied Sciences (2023).

(**C7**) **Heitor R. Guimarães**, Jiaqi Su, Rithesh Kumar, Tiago H Falk, Zeyu Jin. "DiTSE: High-Fidelity Generative Speech Enhancement via Latent Diffusion Transformers." IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) (2026).

(**C8**) **Heitor R. Guimarães**, Ke Tan, Juan Azcarreta, Jesus Alvarez, Prabhav Agrawal, Ashutosh Pandey, Buye Xu. "Improving Resource-Efficient Speech Enhancement via Neural Differentiable DSP Vocoder Refinement." IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) (2025).

(**C9**) Arthur Pimentel, **Heitor R. Guimarães**, Anderson Avila, Mehdi Rezagholizadeh, and Tiago Falk. "How Robust is *Robust wav2vec 2.0* for Edge Applications? An Exploration into the Effects of Quantization and Model Pruning on *In the Wild* Speech Recognition". Edge Intelligence Workshop (2022).

(**C10**) Arthur Pimentel, Yi Zhu, **Heitor R. Guimarães**, and Tiago Falk. "Efficient Audio Deepfake Detection using WavLM with Early Exiting". IEEE International Workshop on Information Forensics and Security (WIFS) (2024).

(**C11**) Yi Zhu, Mahsa Abdollahi, **Heitor R. Guimarães**, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. "Early prediction of honeybee hive winter survivability using

multi-modal sensor data." IEEE International Workshop on Metrology for Agriculture and Forestry (2023).

1.4 Thesis organization

Next, Chapter 2 will present the background material required for the remainder of the thesis. Subsequently, the thesis will be separated into two main parts. The first part focuses on speech processing applications. Within this part, Chapter 3 introduces a methodology to jointly address model compression and robustness to distribution shifts arising from environmental perturbations. Chapter 4 extends this discussion by proposing approaches for designing speech encoders that are both resource-efficient and robust to adversarial attacks. The second part of the thesis then focuses on bioacoustic applications. Chapter 5 explores the use of speech encoders for bioacoustic tasks and introduces a benchmark for acoustic beehive monitoring, where several audio encoders are evaluated. Chapter 6 then presents a novel methodology for learning a resource-efficient bioacoustic encoder that leverages strong DSP-based inductive biases (modulation-spectrum features) to improve generalization in low-capacity models. Finally, Chapter 7 concludes the thesis by summarizing the main contributions and discussing limitations and directions for future research.

Chapter 2

Background

2.1 Deep Neural Networks

As we will describe next, for speech and audio foundation models, two types of neural networks will be central to our discussion. The first is the convolutional neural network (CNN), a specialized architecture designed to process data with a known grid-like topology, such as 1D audio waveforms or 2D spectrograms [25]. CNNs rely on inherent local correlations in the data and are characterized by three fundamental properties: sparse connectivity, parameter sharing, and equivariant representations. These inductive biases allow the network to efficiently capture local, shift-invariant patterns, such as phonetic boundaries or harmonic structures, regardless of their position in the time series.

The second, and currently dominant, architecture is the Transformer [26]. In contrast to the local operations of CNNs, Transformers rely on the Self-Attention mechanism to model global dependencies. By treating the input as a sequence of tokens, Self-Attention allows the model to relate any two positions in the signal directly, regardless of their temporal distance. This capability is crucial for capturing long-range contextual information, which is essential for understanding the high-level semantic structure of speech and audio streams.

The core component of the Transformer architecture is the Multi-Head Self-Attention (MHSA) mechanism. Unlike recurrence or convolution, which process sequences locally or sequentially, Self-Attention allows the model to relate different positions of a sequence to compute a representation of the sequence. Formally, given an input sequence $\mathbf{X} \in \mathbb{R}^{T \times d}$, the mechanism first projects it into

three distinct subspaces: Queries ($\mathbf{Q}$), Keys ($\mathbf{K}$), and Values ($\mathbf{V}$), using learnable weight matrices $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$. The scaled dot-product attention is then computed as:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}, \tag{2.1}$$

where d_k is the dimension of the keys and the term $\sqrt{d_k}$ acts as a scaling factor. The rationale is that for queries and keys initialized with zero mean and unit variance, their dot product has a variance that grows with d_k. Without scaling, for large values of d_k, the dot products can grow large in magnitude, pushing the softmax function into regions with extremely small gradients (i.e., leading to vanishing gradients). To capture diverse contextual information, MHSA performs this operation in parallel across h "heads," allowing the model to attend to information from different representation subspaces at different positions.

A fundamental characteristic of the standard Self-Attention mechanism is its permutation invariance; it treats the input as a set rather than a sequence. Since the order of acoustic frames is critical for speech understanding, the model must be explicitly provided with positional information. Two primary strategies exist for this purpose:

- **Fixed Absolute Positional Encoding**: The original Transformer [26] adds a fixed, non-learnable vector to the input embeddings based on sine and cosine functions of different frequencies. While effective for fixed-length text sequences, this method struggles to generalize to sequence lengths unseen during training [68].
- **Convolutional (Learnable) Positional Embedding**: State-of-the-art speech models [69] adopt a flexible approach tailored for continuous signals. Instead of adding fixed absolute vectors, they employ a convolutional layer placed immediately before the Transformer blocks. By processing the latent features with a specific kernel size and groups, the model implicitly learns relative positional information [70]. While this method introduces additional learnable parameters compared to fixed sinusoidal encodings, it generalizes significantly better to variable-length audio inputs.

2.2 Self-Supervised Learning

2.2.1 Speech Encoders

We begin by examining contrastive pretext tasks to train speech encoders, with Wav2Vec 2.0 [27] serving as the most prominent example. This architecture consists of two primary blocks: a multi-layer convolutional feature encoder that maps raw audio to latent speech representations (typically with a 20 ms stride), and a Transformer-based context network. During pretraining, some latent representations are randomly masked, as in masked language modeling (MLM). This pretraining objective is interesting, as reconstructing the masked segments incentivizes the model to learn rich semantic and paralinguistic representations. The context network then attempts to predict the discrete speech units from a codebook obtained through a Gumbel-Softmax distribution [71], which is selected for its differentiability properties.

Moreover, the model is optimized via a contrastive task. Formally, let $\mathbf{c}_t$ denote the output of the context network at a masked time step t. The contrastive objective forces the model to distinguish the correct quantized target, $\mathbf{q}_t$, from a background of negative examples. We define a candidate set $\tilde{\mathbf{q}} \in \mathbf{Q}_t$ containing $K+1$ latent representations: the true positive $\mathbf{q}_t$ and K distractors sampled from other masked locations. The objective is defined as:

$$\mathcal{L}_m = -\log \frac{\exp(sim(\mathbf{c}_t, \mathbf{q}_t)/\kappa)}{\sum_{\tilde{\mathbf{q}} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \tilde{\mathbf{q}})/\kappa)}, \tag{2.2}$$

where $sim(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T\mathbf{b}/\|\mathbf{a}\|\|\mathbf{b}\|$ denotes the cosine similarity between the context representation $\mathbf{c}_t$ and the quantized targets. Finally, to prevent mode collapse, where the model utilizes only a subset of the codebook entries, an auxiliary codebook diversity loss is jointly minimized. The mathematical definition presented here closely follows the notation from [27], which we refer the interested reader to for a more in-depth discussion.

In contrast to contrastive approaches, predictive pretext tasks for speech representation learning are increasingly being adopted in the field. In this paradigm, the model is trained via direct multi-class classification over a predefined vocabulary, eliminating the need for online quantization or negative sampling. HuBERT [13] and WavLM [28] stand as the current state-of-the-art representatives of

this approach. Architecturally, both models have similar backbones to Wav2Vec 2.0, comprising a convolutional feature encoder followed by a high-capacity Transformer context network.

HuBERT specifically reformulates pretraining as a strict MLM problem. Unlike Wav2Vec 2.0, which learns its targets simultaneously with the encoder, HuBERT utilizes offline clustering to generate stable supervisory signals. Prior to training, a K-means algorithm clusters the acoustic features (e.g., MFCCs) to generate discrete assignments for each frame. The model is then optimized via a BERT-like cross-entropy loss [12] to predict the correct cluster index for masked regions. This mechanism forces the encoder to infer the underlying acoustic and linguistic structure solely from context, effectively serving as a combined acoustic and language model.

Building upon the predictive framework of HuBERT, WavLM [28] extends the architecture to enhance robustness against environmental variability. This improvement is driven primarily by two factors. First, WavLM scales the training corpus to include a massive and diverse collection of audio data, moving beyond the clean audiobook recordings used by its predecessors to incorporate real-world scenarios. Second, the training objective is augmented with a masked speech denoising task. During pretraining, the input utterances are dynamically corrupted via: (i) the simulation of multi-speaker mixtures (overlapping speech), and (ii) the injection of additive noise. Consequently, the model is forced to not only predict the correct discrete units but also implicitly perform speech separation and denoising, thereby learning highly robust acoustic representations.

2.2.2 General-purpose Audio Encoders

Thus far, we have focused on speech encoders, which typically operate as frame-based models: they ingest raw waveforms and output a sequence of latent representations at a high temporal resolution (e.g., 50 Hz). This design is motivated by the nature of speech, where phonetic events occur rapidly (roughly every 20–100 ms), necessitating fine-grained temporal fidelity [72, 73].

In broader audio processing, however, an alternative paradigm known as patch-based modeling has emerged, inspired by Vision Transformers (ViT) [29]. In this approach, the waveform is first converted into a time-frequency representation (e.g., a Mel-spectrogram), which is then partitioned into fixed-size 2D patches. These patches are flattened and processed by a Transformer encoder. While this method often results in lower effective temporal resolution (e.g., 6.25 Hz), smoothing out

short-term dynamics, it excels at capturing global spectral-temporal patterns essential for general audio classification [74].

A prominent example of the patch-based approach is BEATs [30] (and its open-source counterpart, OpenBEATs [75]). During pretraining, BEATs employs a masked modeling objective where a subset of spectrogram patches is masked, and the model predicts corresponding discrete semantic labels. Notably, the training pipeline utilizes an iterative mechanism that alternates between optimizing the encoder and refining the acoustic tokenizer responsible for generating these labels. Empirically, these models demonstrate strong cross-domain generalization, extending effectively to tasks such as bioacoustic monitoring.

Conversely, AVES [31] represents a frame-based adaptation of the speech paradigm for the bioacoustic domain. Architecturally identical to HuBERT, AVES processes raw waveforms via a convolutional front-end to extract features at a 50 Hz frame rate. It is pretrained on a massive mixture of diverse sound events and bioacoustic recordings, allowing it to predict missing segments in an MLM task. This demonstrates that frame-based architectures, while rooted in speech, remain highly effective for general audio and ecological tasks.

Despite their success, both BEATs and AVES rely on relatively high parameter counts (roughly 90M), posing challenges for deployment on resource-constrained edge devices. While smaller self-supervised alternatives exist, such as BYOL-A [76] (5M parameters) and SS-AST [77] (23M), their performance significantly lags behind that of their larger counterparts. This disparity highlights that model capacity remains a critical factor for learning robust representations. Consequently, there is a pressing need for architectures that bridge this gap, retaining the representational power of large-scale SSL pretraining while remaining lightweight enough for real-world IoT deployments.

2.2.3 Assessing the quality of the representations

As universal representations become prevalent, establishing standardized methodologies to compare them is essential. In the speech processing community, the Speech processing Universal PERformance Benchmark (SUPERB) [32] has emerged as the standard leaderboard[1] for evaluating

[1]Recent studies have proposed alternatives to the standard SUPERB framework, specifically investigating how the design and complexity of the probing heads influence the final performance scores. For a detailed analysis of these factors, we refer the reader to [78, 79].

speech foundation models. It assesses performance across ten downstream tasks organized into four categories:

- **Content:** Keyword Spotting (**KS**), Phoneme Recognition (**PR**), Automatic Speech Recognition (**ASR**), and Query-by-Example (**QbE**).
- **Speaker:** Speaker Identification (**SID**), Automatic Speaker Verification (**ASV**), and Speaker Diarization (**SD**).
- **Semantics:** Spoken Intent Classification (**IC**) and Slot Filling (**SF**).
- **Paralinguistics:** Speech Emotion Recognition (**ER**).

For a comprehensive description of these tasks and their datasets, we refer the reader to [32].

Recently, this framework was extended by the SUPERB-Speech Generation (SG) benchmark [33], which introduces generative tasks such as speech translation, voice conversion, and out-of-distribution ASR. In the context of this thesis, we are particularly interested in two tasks highly relevant to "in-the-wild" signal restoration: Speech Enhancement (**SE**) and Source Separation (**SS**).

To quantify the overall capability of a representation, particularly in Chapter 3, we adopt the composite "Score" metric proposed in [80]. This metric normalizes performance across disparate tasks, which may rely on different figures-of-merit (e.g., accuracy vs. equal error rate), into a single scalar value. Formally, the score is defined as:

$$\text{Score}(u) = \frac{1000}{|T|} \sum_{t=1}^{T} \frac{1}{|I_t|} \sum_{i=1}^{I_t} \frac{s_{t,i}(u) - s_{t,i}(\text{base})}{s_{t,i}(\text{SOTA}) - s_{t,i}(\text{base})}, \tag{2.3}$$

where u denotes the upstream model under evaluation (e.g., WavLM), and $s_{t,i}(u)$ represents the value of the i-th metric for task t. The terms $s_{t,i}(\text{SOTA})$ and $s_{t,i}(\text{base})$ correspond to the performance of the state-of-the-art and baseline systems, respectively. $|T|$ is the total number of tasks, and $|I_t|$ is the number of metrics per task. The final score ranges from 0 to 1000, with higher values indicating superior aggregate performance.

Transitioning to the bioacoustic domain, BEANS [34] stands as a prominent benchmark, encompassing 12 tasks: 10 focused on specific bioacoustic challenges and two on general audio events (Sound

Event Classification and Keyword Spotting). The bioacoustic suite covers both classification and detection across diverse species, including datasets for marine mammals, bats, birds, mosquitoes, and dogs, as well as specific detection tasks for endangered species like the Hainan gibbon. Furthermore, as a specific contribution in Chapter 5, we introduce a novel evaluation framework for IoT-enabled beehive monitoring. In Chapter 5, audio encoders are deployed to assess colony health directly from hive recordings. This application is critical; honeybees are foundational to global ecosystems, contributing to an estimated 30% of worldwide food production [63]. Consequently, developing reliable, automated monitoring systems is imperative for ensuring both ecological sustainability and food security.

2.3 Model compression

2.3.1 Quantization and Pruning

Deploying deep neural networks on edge devices presents significant challenges regarding computational overhead, energy consumption, and environmental impact. To mitigate these constraints, we first discuss two primary compression paradigms popular in the literature: quantization [35, 36, 37] and pruning [38, 39, 40].

Quantization minimizes the memory footprint and inference latency by reducing the numerical precision of the model's weights and activations (e.g., converting from 32-bit floating-point to 8-bit integers) [81]. Pruning, conversely, reduces model complexity by identifying and eliminating redundant parameters [41]. This technique is generally categorized into two types:

- **Unstructured Pruning:** Individual connections (weights) are set to zero based on importance metrics, resulting in sparse weight matrices [82]. While theoretically efficient, realizing actual speedups often requires specialized sparse matrix hardware.
- **Structured Pruning:** Entire cohesive groups of parameters (e.g., CNN filters or attention heads) are removed [83, 84]. Unlike its unstructured counterpart, this method directly translates to reduced model size and faster inference on standard hardware, as it preserves the dense matrix structure of the computations.

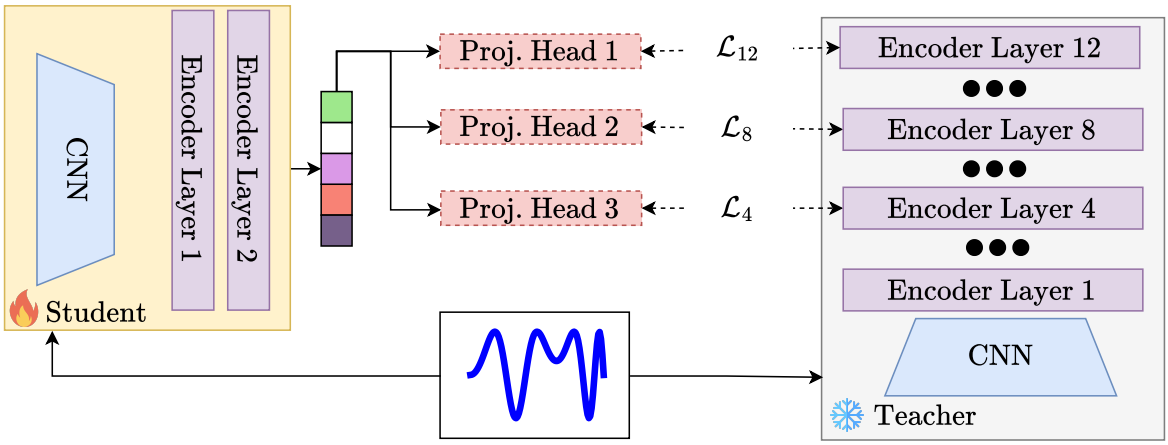


Figure 2.1: Schematic overview of the DistilHuBERT framework. The Teacher model remains frozen throughout the process (indicated by the snowflake icon), while the Student model, initialized with the Teacher's weights, is actively optimized during distillation (indicated by the flame icon).

2.3.2 Knowledge Distillation

The primary objective of Knowledge Distillation (KD), originally proposed by [42], is to transfer knowledge from a high-capacity Teacher model, T_θ, to a compact Student model, S_ϕ, thereby enabling the smaller network to approximate the generalization capabilities of the larger one. In a standard supervised setting, KD is typically formulated as minimizing a composite objective:

$$\min_{\phi} (1-\alpha)\mathcal{L}\mathrm{CE}(S_\phi(x), y) + \alpha D_{\mathrm{KL}}(S_\phi(x), T_\theta(x)), \tag{2.4}$$

where $\mathcal{L}_{\mathrm{CE}}$ denotes the cross-entropy loss with respect to the true labels y, D_{KL} is the Kullback–Leibler divergence between the Teacher and Student output distributions (logits), and α is a hyperparameter balancing the two terms.

While classical KD focuses on output probabilities, self-supervised speech encoders, which often lack explicit class labels, require feature-based distillation. Techniques for distilling these representations have been extensively explored in recent literature [43, 44].

As illustrated in Figure 2.1, DistilHuBERT [45], for instance, leverages this paradigm to compress the HuBERT model by approximately 75%. Specifically, it employs a Student model consisting of only two Transformer layers. The Student's output is projected via three separate prediction heads to approximate the latent representations of the Teacher's 4^{th}, 8^{th}, and 12^{th} layers. DistilHuBERT

optimizes a combination of an L_1 loss and a cosine similarity term to align both the magnitude and orientation of the representations.

Formally, let $\hat{\mathbf{z}}_t^{(l)} \in \mathbb{R}^D$ denote the projected Student representation and $\mathbf{z}_t^{(l)} \in \mathbb{R}^D$ the corresponding Teacher representation at layer l and time frame t. The objective is defined as:

$$\mathcal{L} = \frac{1}{T|L|} \sum_{t=1}^{T} \sum_{l \in L} \left(\frac{1}{D} \left\| \hat{\mathbf{z}}^{(l)} t - \mathbf{z}^{(l)} t \right\|_1 - \log \sigma \left[\cos \left(\hat{\mathbf{z}}^{(l)} t, \mathbf{z}^{(l)} t \right) \right] \right), \tag{2.5}$$

where $L = \{4, 8, 12\}$ is the set of target Teacher layers, T is the total number of frames, $\sigma(\cdot)$ is the sigmoid function, and $\cos(\cdot, \cdot)$ denotes cosine similarity.

Due to its effectiveness and simplicity, DistilHuBERT has served as a foundation for subsequent improvements. Concurrently, FitHuBERT [85] and the Deep-versus-Wide (DW) [86] frameworks proposed architectural modifications to the Student. Both approaches argue for a "deeper and thinner" Student architecture rather than DistilHuBERT's "shallow and wide" design. Furthermore, the DW method explores layer-to-layer (L2L) mapping, where intermediate Student layers are aligned with corresponding Teacher layers, unlike DistilHuBERT, which predicts multiple Teacher layers solely from the Student's final output. Empirically, these deeper distilled models demonstrate superior fidelity to the Teacher representations.

LightHuBERT [46] employs a two-stage distillation mechanism coupled with structured pruning. While currently achieving state-of-the-art compression results for speech encoders, its training pipeline is computationally intensive, often proving prohibitive for academic-scale resources.

More recently, DPHuBERT [47] introduced a joint distillation and pruning framework designed to optimize performance on the SUPERB benchmark. The training proceeds in two distinct phases: first, a Student model, initialized as a copy of the Teacher, is trained with sparsity-inducing regularization, effectively eliminating 75% of parameters via structured pruning [87]. In the second phase, a final distillation step fine-tunes the remaining weights to maximize the similarity between the compressed Student and the original Teacher.

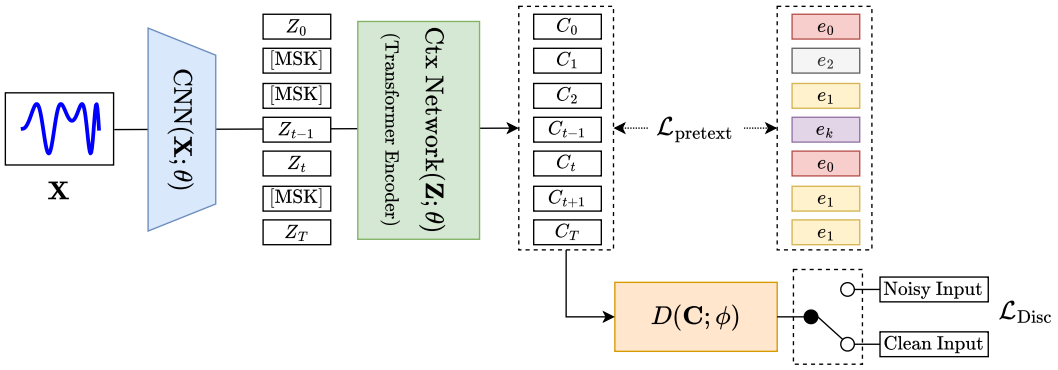


Figure 2.2: Schematic overview of the Domain Adversarial Training (DAT) framework within Robust HuBERT. A discriminator network, $D(\cdot;\phi)$, processes the latent representations from the context network and is trained to distinguish between clean and noisy inputs. Simultaneously, the encoder is optimized to generate noise-invariant representations designed to fool the discriminator, thereby enforcing robustness against environmental perturbations.

2.4 Domain Adaptation

For speech encoders, models are still sensitive to conditions typically observed in edge applications. For example, recent research has shown that when test data is corrupted by factors not seen during training (e.g., reverberation and noise), the distribution shift hampers model performance [48]. This has been shown to be common across speech representations, as they are usually trained on extensive unlabelled (clean) data from audiobook readings [88, 89]. Techniques that have been explored to mitigate this out-of-domain issue include domain adaptation and domain generalization [49]. Robust HuBERT [50], for example, proposes to use domain adversarial training (DAT) [51] to learn noise-invariant features, using the HuBERT model as a baseline, and improved robustness to noisy test conditions have been reported [50]. Figure 2.2 depicts a simplified overview of the Robust HuBERT framework.

As can be seen, the methods discussed above either focus on improving the robustness of universal representations to unseen conditions or on enabling the compression of model sizes. To the best of our knowledge, only one work [90], developed concurrently with ours, has investigated the compression and robustness of speech foundation models. Inspired by the Robust HuBERT model, the authors combine domain adversarial training (DAT) and the bootstraping-your-own-latents (BYOL) [91] style of training. Specifically, the Teacher and Student networks receive different distorted views of the same utterance as input. Unlike our work, the HuBERT model is distilled and evaluated only on three downstream tasks. We also encountered three key differences from our work: (i) We learn

noise-invariance via a multi-task learning approach instead of DAT; (ii) we show that our recipe can be applied to other Teacher models beyond just HuBERT; and (iii) we extend our analysis to all ten tasks from the SUPERB benchmark and two tasks from the SUPERB-SG benchmark. Our approach, termed RobustDistiller, extends both the DistilHuBERT and DPHuBERT recipes to generate robust universal representations and will be discussed in greater detail in Chapter 3.

2.5 Adversarial Attacks

2.5.1 Formal definition and Adversarial Training

In a standard classification setting, let $f_\theta : \mathbb{R}^D \mapsto \mathbb{R}^K$ denote a classifier parametrized by θ, where D is the input dimension and K is the number of target classes. Let $l : \mathbb{R}^K \times \mathbb{Z}_+ \mapsto \mathbb{R}_+$ be a loss function. From the perspective of statistical learning theory [92], the *empirical risk* of a classifier is defined as the expected loss over a finite set of i.i.d. samples, $\mathcal{D}_{\text{train}} = \{x_i \in \mathbb{R}^D, y_i \in \mathbb{Z}_+\}_{i=1,\dots,m}$, given by:

$$\hat{R}(f_\theta, \mathcal{D}_{\text{train}}) = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{train}}} l(f_\theta(x), y). \tag{2.6}$$

While the objective of standard machine learning algorithms is to minimize this empirical risk (i.e., finding the optimal parameters θ), adversarial attacks seek to maximize the loss function. An adversary constructs a perturbation δ, constrained within a p-norm ball in the input metric space, to induce misclassification. Analogously, we define the empirical adversarial risk [93] as:

$$\hat{R}_{\text{adv}}(f_\theta, \mathcal{D}_{\text{train}}) = \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{train}}} \max_{\delta\in\Delta(x)} l(f_\theta(x+\delta), y), \tag{2.7}$$

where $\Delta(x) = \{\delta : ||\delta||_p \leq \epsilon\}$ represents the set of allowable perturbations.

Furthermore, adversarial attacks are commonly categorized according to the amount of information available to the attacker. In white-box attacks, the adversary has full knowledge of the target model, including access to its architecture and internal gradients. In contrast, black-box attacks assume that the adversary can only query the model and observe its outputs, without access to its internal representations or gradients.

To defend against such threats, the most prominent strategy is adversarial training (AT) [52]. AT formulates the learning process as a min-max robust optimization problem, where the goal is to minimize the worst-case loss generated by the adversary:

$$\min_{\theta} \hat{R}_{\text{adv}}(f_\theta, \mathcal{D}_{\text{train}}) = \min_{\theta} \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(x,y)\sim\mathcal{D}_{\text{train}}} \max_{\delta\in\Delta(x)} l(f_\theta(x+\delta), y). \tag{2.8}$$

In practice, AT functions as a rigorous data augmentation technique, where the model is continuously trained on adversarial versions of natural samples generated dynamically during optimization.

2.5.2 Adversarial Robustness in Audio Encoders

While extensive research has focused on increasing the robustness of speech encoders against environmental noise, significantly fewer studies have addressed adversarial robustness. Wu et al. [53] investigated the vulnerability of prominent speech encoders, such as Wav2Vec 2.0 and HuBERT, demonstrating their susceptibility to attacks like the Basic Iterative Method (BIM). They further analyzed the impact of limiting the attacker's knowledge of the model. In this context, grey-box attacks are typically defined as scenarios where the adversary has full access to the upstream encoder but no knowledge of the downstream classification head. Conversely, black-box attacks involve generating perturbations using a surrogate model and transferring them to the target system.

Furthermore, Guimarães et al. [94] examined the adversarial robustness of keyword spotting systems built upon these encoders. This work explored a broader range of upstream models and attack methods, and also provided an initial analysis of the spectral characteristics of adversarial perturbations in the modulation frequency domain [95]. Both studies highlight a critical security gap: current state-of-the-art speech encoders remain inherently fragile to malicious interference, necessitating the development of certified defense mechanisms.

2.5.3 Robust Knowledge Distillation

While Knowledge Distillation is effective for model compression, it does not inherently preserve the Teacher's adversarial robustness, as demonstrated by Goldblum et al. [54]. To mitigate this, Adversarially Robust Distillation (ARD) [54] was proposed as a framework that integrates adversarial

training directly into the distillation objective. The core principle is to force the Student S_ϕ, when exposed to adversarial perturbations, to align its output with that of the Teacher T_θ on the corresponding clean input. The objective function is formulated as:

$$\min_{\phi} (1-\alpha)\mathcal{L}_{\mathrm{CE}}(S_\phi(x), y) + \alpha D_{\mathrm{KL}}(S_\phi(x+\delta), T_\theta(x)). \tag{2.9}$$

By minimizing the Kullback–Leibler divergence between the logits of the robust Teacher (on clean data) and the Student (on adversarial data), the Student learns to become invariant to the perturbations δ.

Building on this, Robust Soft Label Adversarial Distillation (RSLAD) [55] emerged as an alternative strategy. RSLAD argues that replacing ground-truth hard labels with the Teacher's soft labels provides richer guidance during adversarial training, empirically yielding superior robustness compared to ARD. However, as we will demonstrate in Chapter 4, neither method consistently outperforms a compact model trained solely via standard Adversarial Training in the context of self-supervised speech representations. This observation suggests that existing robust distillation recipes are insufficient for current audio foundation models, highlighting the need for specialized techniques tailored to the geometric properties of high-dimensional speech embeddings.

Part I

Applications to Speech Encoders

Chapter 3

Noise-invariant features via denoising distillation and multi-task learning

3.1 Preamble

This chapter is compiled from material extracted from the following publications: Our extended abstract at the 2022 Edge Intelligence Workshop [96], our conference paper at the Efficient Natural Language and Speech Processing (ENLSP-II) Workshop held at the 2022 Advances in Neural Information Processing Systems (NeurIPS) [97], and the conference paper at the 2023 IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) [48].

3.2 Introduction

As discussed in Chapter 2, speech foundation models are crucial for edge applications, where storage and computational capacity are inherently limited. However, deploying universal speech representations in resource-constrained environments presents two fundamental challenges. First, their performance degrades significantly under distribution shifts (e.g., environmental noise and reverberation) not encountered during training [50]. Second, these models typically require large parameter counts, making them impractical for low-resource devices. Notwithstanding, recent findings indicate a critical relationship between model capacity and robustness: compressed models

are disproportionately sensitive to unseen environmental conditions compared to their full-sized counterparts [96].

While recent compression frameworks involving joint distillation and pruning (e.g., LightHuBERT and DPWavLM) have demonstrated success in clean settings, their efficacy in noisy "in-the-wild" scenarios remains limited. Although these methods can narrow the performance gap between Student and Teacher models in the presence of noise, significant degradation persists, particularly in speaker-related downstream tasks. As such, only a handful of works have successfully combined compression and environmental robustness for edge applications. This chapter aims to fill this gap.

To this end, we propose RobustDistiller, a novel training framework designed to compress universal speech encoders while simultaneously enhancing their resilience to unseen noise perturbations. Our main contributions are summarized as follows:

1. We propose a Feature Denoising Knowledge Distillation framework that enables the Student model to learn noise-invariant representations, improving robustness in noisy environments.
2. Through extensive experimentation, we demonstrate that the denoising objective yields better feature disentanglement in the Student model, thereby improving generalization.
3. We validate the proposed methodology across multiple upstream encoders (HuBERT, Wav2Vec 2.0, WavLM, and Robust HuBERT), confirming its task- and model-agnostic nature.
4. Empirically, we show that our distilled Student can outperform its robust Teacher in noisy scenarios for content-related tasks, particularly in low-SNR conditions (-5 to 5 dB).
5. Finally, while our primary implementation is built upon the DistilHuBERT architecture, we demonstrate that the framework is flexible and complementary to other compression strategies, such as the DPWavLM.

3.3 RobustDistiller: Joint distillation and noise-invariant representation extraction

The RobustDistiller recipe, as introduced in [48], was developed to compress the HuBERT-*base* encoder model to a quarter of its original size, thus reducing the number of parameters from 94

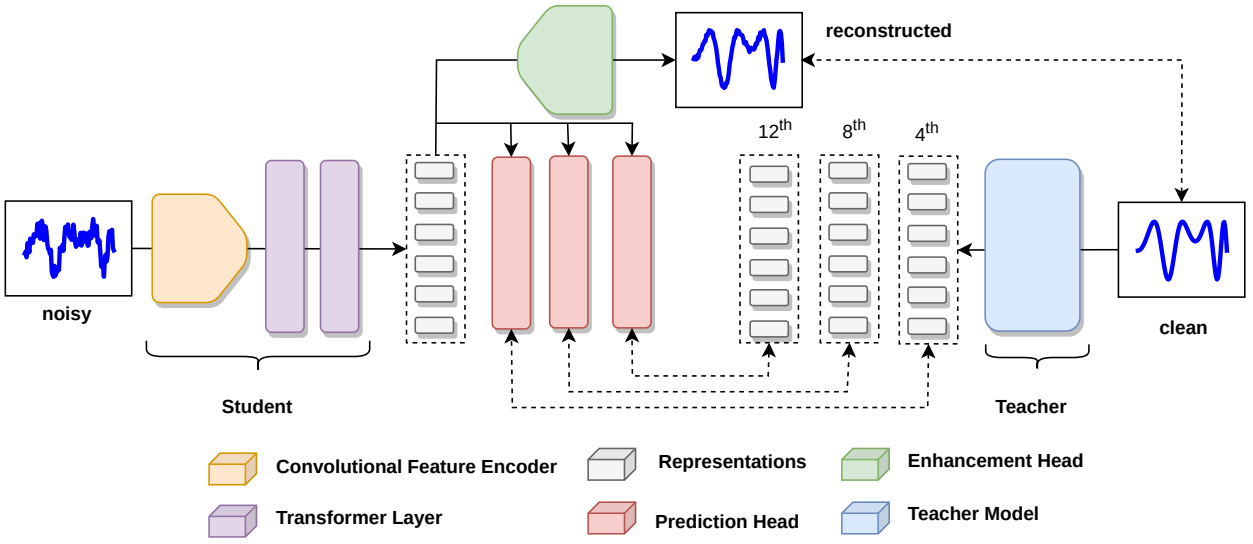


Figure 3.1: Diagram of the proposed RobustDistiller recipe. The input of the Student model can be a noisy waveform. The Teacher, on the other hand, receives the clean waveform. The Student must learn clean features and simultaneously retrieve as much information as possible to reconstruct a clean utterance.

million to 23 million approximately. This was achieved while creating a more robust model against environmental artifacts. Initially, we proposed RobustDistiller as an extension of the DistilHuBERT recipe, guided by a robustness-first principle. Herein, we show that our proposed recipe can be extended to other distillation recipes, as discussed and empirically shown in the following sections. Two main modifications are proposed to improve the robustness of the Student models: (i) a Feature Denoising Knowledge Distillation step that induces the Student model to learn noise-invariant representations; and (ii) a multi-task learning approach via a signal enhancement step where, given the last hidden state from the Student model obtained from a noisy signal, we reconstruct the clean waveform or estimate a binary mask to be applied on the STFT of the noisy input. Figure 3.1 shows the schematics of the proposed model. More details about these modifications are given next.

3.3.1 Modification #1: Feature Denoising Knowledge Distillation

As stated earlier, one of the main challenges that speech foundation models face is domain shift, where the training and real-world test data differ significantly, e.g., in channel conditions and environmental noises. This can degrade system performance, rendering the models less effective for practical use. Data augmentation has been proposed to address this issue by increasing data diversity and reducing model bias [98].

Inspired by the WavLM training scheme, here we introduce an online contamination method during the distillation process. Our method involves training a Student model to reconstruct the clean representations of a Teacher model while being fed with noisy data as input. During training, a batch of clean speech utterances is uniformly sampled and subjected to one of the following actions: (i) no changes are made to the training utterance; (ii) the utterance is contaminated with additive noise with a signal-to-noise ratio (SNR) chosen randomly from $[0, 20]$ dB; (iii) the speech waveform is convolved with a random room impulse response; and (iv) both noise and reverberation are applied simultaneously. Mathematically, the goal of feature denoising knowledge distillation is to minimize the distance between the Student and Teacher's representation, as follows:

$$\min_{\theta} \; \mathbb{E}_{x\sim\mathcal{D}} \left[\frac{1}{m} \sum_{i=1}^{m} \sum_{\ell \in L} \mathcal{L}_D \left(f_T^{(\ell)}(\boldsymbol{x}_i; \phi), f_S^{(\ell)}(\hat{\boldsymbol{x}}_i; \theta) \right) \right], \tag{3.1}$$

where $\boldsymbol{x}_i$ is sample from an unlabeled dataset $\mathcal{D} = \{x_i \mid i = 1, ..., m\}$ and $\hat{\boldsymbol{x}}_i$ is the distorted view of such sample. $f_T^{(\ell)}$ is the hidden representation of the ℓ-th layer from the Teacher model parametrized by ϕ, and $f_S^{(\ell)}$ is the Student prediction that approximates the Teacher representation. L is the set of layers chosen to be distilled (e.g., 4^{th}, 8^{th}, and 12^{th} if following the DistilHuBERT recipe). Lastly, $\mathcal{L}_D$ follows the same definition of Equation 2.5 as a combination of L1 loss and a cosine similarity between representations.

With this feature denoising step, our central hypothesis is that the model must learn to disentangle the speech signal information from the background noise, leveraging more robust (noise-invariant) features. Note that this methodology can be applied to both layer-wise distillation and L2L methods.

3.3.2 Modification #2: Multi-task learning

Speech enhancement is a fundamental task in speech processing aimed at improving the quality of a speech signal by removing detrimental factors. This thesis proposes a multi-task learning approach inspired by speech enhancement and motivated by recent advancements in speech representation learning [56]. In this setting, our proposed approach involves reconstructing the clean waveform from the learned representation of the Student model via an additional enhancement head in complement to the feature denoising knowledge distillation process. Unlike traditional enhancement techniques,

we aim to ensure that the upstream model carries sufficient information about the speech signal rather than focusing on optimizing a quality metric.

To achieve this goal, we explored two architectures for the enhancement head. First, we experiment with an enhancement head similar to the encoder of the Student model. Specifically, we employ a BiLSTM layer followed by seven transposed convolutions and GELU activation functions. This method aims to reconstruct the waveform directly from the last hidden state of our Student model. The main limitation of this proposed head is that the enhancement head is large, with roughly 17 million parameters. Although we will discard the enhancement head after training, it can be a memory bottleneck during training.

Next, we also experiment with a lightweight enhancement head that operates in the time-frequency domain. This enhancement head has only 5 million parameters, not impacting the original distillation time too much. Herein, we estimate a binary mask that will be applied to the magnitude of the noisy signal's short-time Fourier transform (STFT). Later, we reconstruct the clean waveform to measure the quality of the enhanced signal (e.g., PESQ [99]) by using the noisy phase of the signal. This enhancement head consists of a single BiLSTM network with three layers and a sigmoid activation function to create the binary mask.

Additionally, we experiment with three different loss functions: L1-, L2-, and a multi-resolution STFT-loss to guide the enhancement head. In our experiments, we will discuss the impact of each choice in more detail. We refer to the combination of these two procedures, i.e., feature denoising knowledge distillation and multi-task learning, as the RobustDistiller recipe, which can be applied across unsupervised representations. In our experiments, we build on the work of [48] and apply the RobustDistiller recipe to distill four popular speech representations: *Wav2Vec 2.0* [27], HuBERT [100], WavLM [28], as well as the more recent Robust HuBERT[50].

3.3.3 Modification #3: (Optional) Curriculum learning

Inspired by the ideas of curriculum learning [101], we propose a progressive training mechanism where the network gradually receives more deteriorated speech samples during training iterations and not randomly from the beginning of the optimization process. Based on the literature, we expect the models trained on this regime to have better generalization capabilities, which can lead us to

reinforce the model behavior in learning speech features disentangled from noise. In our work, we gradually increase the difficulty of the online contamination process by defining a custom sampling procedure instead of allowing every scenario from the beginning. This process is done until we reach half of the training steps, and after, the same regime of the previous section is applied. Algorithm 1 shows the pseudocode to describe the progressive learning strategy adopted in this work. Note that, if $\tau = 0$ and $t = 1$, we recover the augmentation strategy, Modification #1, without curriculum learning.

Algorithm 1: Sample strategy for SNR-Progressive Learning

given a batch of utterances $U = \{u_i\}_{i=0}^{L}$, a set of noise files U_n, a set of room impulse responses U_r, the current iteration it, and the total number of iterations N ;
for *each utterance* u_i **do**
  a ← sample an action from the equiprobable sample space $\{a1,\ a2,\ a3,\ a4\}$;
  if $a == a2$ **then**
    $\tau \leftarrow 20\left(1 - \frac{2it}{N}\right)$ **if** $it \leq N/2$ **else** 0 ; `/* Lower-bound for the SNR value */`
    $snr \leftarrow$ sample an integer number from a uniform distribution $\mathcal{U}(\tau, 20)$;
    $p_n \leftarrow$ sample a random number from a uniform distribution $\mathcal{U}(0, 1)$;
    $u_n \leftarrow$ sample from U_n **if** $p_n \leq 0.7$ **else** return a narrow-band white noise $\mathcal{N}(0, 1)$;
    $u_i \leftarrow$ applyNoise(u_i, u_n, snr)
  else if $a == a3$ **then**
    $t \leftarrow \frac{2it}{N}$ **if** $it \leq N/2$ **else** 1 ; `/* Threshold for applying reverberation */`
    $p_r \leftarrow$ sample a random number from a uniform distribution $\mathcal{U}(0, 1)$;
    if $p_r \leq t$ **then**
      $u_r \leftarrow$ uniformly sample a RIR from U_r ;
      $u_i \leftarrow$ applyReverberation(u_i, u_r) ;
    end
  else if $a == a4$ **then**
    $u_i \leftarrow$ apply the same mechanism from the action $a2$;
    $u_i \leftarrow$ apply the same mechanism from the action $a3$;
  else
    Do nothing ;
  end
end

3.3.4 Notes on extending RobustDistiller to DP-methods

It is interesting to notice that, although RobustDistiller's recipe was originally developed on top of the DistilHuBERT method, we can expand this idea to other methods as well. To test this hypothesis, in this work we explore expanding the DPWavLM model with our modifications. More specifically, our proposed modifications are inserted during the second distillation step of the recipe, after the

distillation-plus-pruning step. We opted for this implementation to reduce the computational cost; however, it is an interesting future work to explore the effects of robust distillation on both steps.

3.4 Experimental Setup

3.4.1 Datasets

Similar to DistilHuBERT, the distillation recipe of the proposed model is based on the clean speech utterances of the LibriSpeech corpus [88], a dataset with 960 hours of audiobook recordings with a 16 kHz sampling rate derived from the LibriVox project. Additionally, to augment the training data, two noise datasets are used, namely MUSAN [102] and UrbanSound8K [103], as well as the OpenSLR28 dataset [104] of room impulse responses (RIR). The noise datasets contain approximately 15 hours of recordings in a wide variety of categories, from office-like noises, natural sounds, babble noise, and urban sounds, such as engine idling and siren. We removed the children playing and street music categories to focus on non-speech-like noise sources in this first analysis. All the utterances are resampled to 16 kHz. Lastly, the OpenSLR28 dataset contains 325 RIRs recorded in real environments, from small meeting rooms to large auditoriums.

At test time, two additional noise datasets are used to test the model performance under unseen conditions. The first is the Acoustic Scene Classification from the Detection and Classification of Acoustic Scenes and Events Challenge (DCASE2020) [105], comprised of 64 hours of audio recordings in 10 acoustic scenes, with four different recording devices in 12 cities. For the reverberation condition, we used the OpenSLR26 dataset [104], which contains 60,000 simulated room impulse responses corresponding to various small-, medium-, and large-sized rooms.

3.4.2 Pre-training

Here, we evaluate the effectiveness of the proposed methodology through a series of experiments. For the distillation process, we selected four Teacher models: *Wav2Vec 2.0* base, HuBERT base, WavLM base+, and Robust HuBERT. The distillation process, as outlined in [45], was applied to these models, resulting in models with the *Distiller* prefix. We should note that the distillation process was applied to the 4^{th}, 8^{th}, and 12^{th} layers of all Teacher models, which may not be optimal for models

other than HuBERT [45]. For those interested in achieving maximum task-based performance with distilled speech encoders, refer to [106]. Therefore, the reported results may underestimate the achievable accuracy. In the next section, we also show the experimental results of applying our recipe in the DPWavLM method. We follow closely the original procedure described in its original paper.

For all experiments derived from the DistilHuBERT recipe, upstream models were trained using a single NVidia A100 GPU. When utilizing feature denoising KD only, the proposed method requires approximately 30 hours to train, whereas adding the waveform denoising head increases training time to approximately 43 hours. For the experiments based on the DPWavLM recipe, we follow the configurations on the original paper, training with 4 NVidia A100 GPU for approximately 8h. Unless stated otherwise, the reported results were achieved using the feature denoising KD and the STFT-denoising head with the L1 loss. We employed the AdamW optimizer with a batch size of 24 utterances for 200k iterations, and the learning rate linearly decays from 2×10^{-4} to zero after 14k updates.

3.4.3 Downstream Tasks, Testing conditions, and Figures-of-Merit

The ten original tasks from the SUPERB benchmark [32] and two from the SUPERB-SG benchmark [33] are used to gauge the benefits of the proposed method. It is important to emphasize that since our focus is on the upstream model, we do not perform hyperparameter-tuning on the SUPERB downstream tasks; hence we use the default parameters reported in the SUPERB benchmark. Consequently, for the clean test scenario, some of the results reported herein for the baselines may differ from those reported on the SUPERB benchmark. Nonetheless, the performance trend and gaps reported here are similar to those seen on the leaderboard.

Four evaluation scenarios are considered for analysis for all downstream tasks in our work: clean (c), noise-only (n), reverberation-only (r), and noise-plus-reverberation ($n + r$). The clean scenario meets the same criteria as the test set as SUPERB downstream tasks. For the noise-only setting, additive noise with signal-to-noise ratios ranging from $[-5, 20]$ dB is sampled from the DCASE2020 dataset and added to the utterances. For the reverberation-only condition, in turn, room impulse responses are uniformly sampled from the OpenSLR26 dataset and convolved with the test set utterance. Lastly, the noise-plus-reverberation condition applies both noise and reverberation jointly as described in the (n) and (r) settings, thus representing the most challenging configuration. A

fixed custom seed is applied for the random processes to ensure all models are evaluated with the same deterioration.

As for figures-of-merit, each downstream task has its own evaluation metric. For the classification tasks, i.e., KS, SID, IC, and ER, the standard accuracy metric (ACC) is reported. For the SF task, both F-score and Character Error Rate (CER) are used. Word-Error rate (WER), Phoneme-Error rate (PER), and Diarization-Error rate (DER) are the metrics used for the ASR, PR, and SD tasks, respectively. For the ASV task, the equal-error rate (EER) is used. For the QbE task, we report the maximum term weighted value (MTWV) metric. Next, the SE task is evaluated using quality and intelligibility metrics: the Perceptual Evaluation of Speech Quality (PESQ) [99], the Short-Time Objective Intelligibility (STOI) [107], and the Scale-Invariant Signal-to-Distortion Ratio (SI-SDR) [108]. The SS task is also evaluated using the SI-SDR metric. More details about the figures-of-merit can be found in [32, 33]. Lastly, we also adopt the "score" metric defined in Equation 2.3.

3.5 Experimental Results and Discussion

3.5.1 Performance across SUPERB tasks

In this section, we report the accuracy achieved across the 10 SUPERB tasks with the four baseline Teacher models: Wav2Vec 2.0 base, HuBERT base, WavLM base, and Robust HuBERT, each with 95 million parameters. Moreover, we additionally show the results achieved with different compression recipes, namely *LightHuBERT* (27M parameters), four distilled versions of the Teacher models using the DistilHuBERT recipe (23M parameters, each), the FitHuBERT model (23M parameters), and the DPHuBERT and DPWavLM (24M parameters, each). Finally, we report the performance achieved with the proposed RobustDistiller recipe for the four Teacher models, represented by the acronym *RD* in the tables. The results are shown for the clean test conditions (column (c) in the tables), noise-only (n), reverberation-only (r), and noise-plus-reverberation (n+r) conditions. The performance metric used by each task is displayed, as well as an up- or down-arrow indicating if higher or lower metric values are better, respectively.

Table 3.1: Experimental results for content-related tasks on the SUPERB. Bold represents the best result and underline the second best for the compressed models.

Upstream	**KS** - ACC (↑)				**PR** - PER (↓)				**ASR** - WER (↓)				**QbE** - MTWV (↑)			
	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)
Teachers																
Wav2Vec 2.0 Base [27]	96.20	86.50	64.43	56.25	5.48	17.61	18.86	36.59	6.44	24.55	27.26	56.06	8.44	7.25	7.33	6.20
HuBERT Base [100]	96.30	84.55	61.70	51.61	5.05	15.07	14.41	31.12	6.43	20.12	19.26	42.47	6.52	6.54	5.53	5.15
WavLM Base+ [28]	96.88	90.07	78.64	72.38	3.93	11.33	10.83	30.63	5.87	14.36	13.94	39.17	6.25	6.02	6.16	5.29
Robust HuBERT [50]	96.33	92.15	74.81	66.50	5.38	9.84	13.92	28.38	6.75	14.74	21.55	41.73	5.50	5.92	5.83	5.04
Distillation-only recipes																
Distiller (Wav2Vec 2.0) [45]	95.33	82.21	55.18	45.37	15.70	37.65	67.54	77.62	13.00	46.89	113.63	181.22	6.56	4.80	5.00	5.17
Distiller (HuBERT) [45]	96.14	86.34	59.20	54.53	14.15	32.95	51.40	66.87	13.26	40.55	72.07	112.73	6.72	6.14	4.99	4.44
Distiller (WavLM) [45]	96.36	88.48	61.34	57.58	**11.57**	26.70	49.87	64.19	12.30	37.40	91.46	153.59	6.17	6.34	5.80	4.49
Distiller (Robust HuBERT) [45]	96.62	87.76	58.65	50.70	12.42	28.57	56.10	71.54	12.69	36.27	91.28	145.56	6.53	6.02	4.59	4.18
FitHuBERT [85]	96.59	89.78	61.31	56.77	13.07	27.32	38.45	55.30	**11.56**	34.20	54.54	101.33	6.21	5.79	5.06	4.38
[**Ours**] RD (Wav2Vec 2.0)	96.14	91.98	85.49	78.94	16.49	24.86	26.77	38.82	13.53	25.19	27.24	43.36	6.27	6.12	5.89	4.78
[**Ours**] RD (HuBERT)	96.07	92.31	85.43	79.32	15.07	22.71	23.74	35.16	13.37	24.44	25.60	40.81	**6.82**	**7.35**	**6.94**	**6.02**
[**Ours**] RD (WavLM)	**96.79**	**93.48**	**87.76**	**82.31**	13.38	20.68	**21.43**	32.49	13.02	**23.75**	**24.46**	**39.73**	5.89	6.03	5.99	5.44
[**Ours**] RD (Robust HuBERT)	96.56	92.99	85.69	79.81	13.53	**20.58**	21.50	**32.38**	13.17	23.97	25.31	40.85	6.61	6.57	6.66	5.19
Distillation-*plus*-Pruning recipes																
LightHuBERT [46]	96.11	86.69	72.35	62.25	**6.14**	16.92	**12.15**	27.42	**8.22**	25.33	20.30	43.80	5.90	**6.17**	**6.22**	4.90
DPHuBERT [47]	96.30	84.49	63.84	56.05	9.41	24.28	29.40	49.03	10.60	32.76	40.12	76.46	**6.28**	5.91	4.96	**5.13**
DPWavLM [47]	96.20	88.12	60.40	55.08	8.23	18.81	25.95	43.78	10.14	26.17	37.20	67.93	6.04	5.46	5.20	5.05
[**Ours**] RD (DPWavLM)	**96.69**	**92.66**	**88.96**	**83.93**	8.26	**13.19**	12.87	**21.44**	9.98	**18.37**	**18.34**	**31.59**	6.00	6.04	5.85	5.00

Robustness of Content-related Tasks

First, we start looking at the robustness experiments, which constitute the central part of our contribution. Table 3.1 shows the obtained results for the four content-related tasks. As can be seen, for the KS task, the *WavLM* model showed the best performance among the Teacher models for three of the four conditions, with Robust HuBERT showing the highest accuracy for the noise-only condition. In the clean and noise-only conditions, the compressed benchmarks performed similarly to the Teacher models; however, the performance degraded substantially under reverberation-only and reverberation-plus-noise settings. The proposed methodology outperformed the original Distiller recipe across all scenarios, including clean conditions. In the challenging noisy conditions, the RobustDistiller framework allowed for compressed models to even outperform their larger Teacher versions, thus showing the advantage of the feature denoising KD and multi-task learning for speech representation learning and compression.

For the PR and ASR tasks, WavLM and Robust HuBERT continued to show the best accuracy across the Teachers, but with substantial performance drops in the three noisy conditions. For these two tasks, unlike KS, the Distiller compression methodology highly degraded performance with FitHuBERT achieving the top performance on the clean tested conditions. The proposed Robust-Distiller framework was able to outperform the original distillation methodology and FitHuBERT on noisy conditions, but not the recipes based on distillation-plus-pruning (e.g., DPWavLM and

Table 3.2: Experimental results for speaker-related tasks on the SUPERB. Bold represents the best result and underline the second best for the compressed models.

Upstream	**SID** - ACC (↑)				**ASV** - EER (↓)				**SD** - DER (↓)			
	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)
Teachers												
Wav2Vec 2.0 Base [27]	75.17	41.79	5.36	2.80	6.01	15.33	28.45	34.05	6.08	8.33	40.44	46.91
HuBERT Base [100]	81.43	52.44	12.37	6.46	5.50	12.64	24.41	28.80	5.88	8.02	28.51	32.55
WavLM Base+ [28]	72.14	49.52	14.08	6.74	4.34	10.16	20.88	28.90	3.58	5.50	16.25	21.11
Robust HuBERT [50]	73.08	48.43	6.99	4.24	5.26	10.05	26.01	31.28	4.79	6.21	19.51	22.49
Distillation-only recipes												
Distiller (Wav2Vec 2.0) [45]	73.63	41.17	3.07	1.49	7.27	17.35	39.18	41.90	6.59	8.31	59.72	65.24
Distiller (HuBERT) [45]	72.28	45.01	5.11	2.95	6.59	13.94	29.32	33.84	5.98	7.95	39.51	41.98
Distiller (WavLM) [45]	**79.78**	52.19	6.07	3.51	**5.69**	11.51	33.02	35.82	5.75	7.74	53.52	60.11
Distiller (Robust HuBERT) [45]	77.37	50.67	5.15	3.07	6.08	12.08	36.90	40.01	6.12	7.71	42.67	49.27
FitHuBERT [85]	54.09	33.04	7.22	4.04	6.72	13.86	33.49	37.75	6.84	8.99	33.79	40.63
[**Ours**] RD (Wav2Vec 2.0)	76.27	61.06	40.18	28.86	7.31	10.69	13.51	16.89	5.22	6.99	7.69	**9.39**
[**Ours**] RD (HuBERT)	77.91	61.83	**43.56**	**31.66**	6.23	9.23	11.66	15.01	5.19	6.40	**8.01**	10.46
[**Ours**] RD (WavLM)	78.39	60.73	42.04	30.02	5.83	**8.93**	**11.24**	**14.59**	**4.92**	**6.28**	9.01	12.09
[**Ours**] RD (Robust HuBERT)	77.52	**61.96**	41.79	29.66	6.59	9.64	12.22	16.02	5.31	6.53	8.15	10.63
Distillation-*plus*-Pruning recipes												
LightHuBERT [46]	72.03	41.57	17.50	8.68	5.57	12.80	14.81	23.56	6.08	7.96	19.38	21.73
DPHuBERT [47]	76.33	45.79	8.13	4.27	5.54	12.93	23.93	30.80	6.16	7.97	26.18	32.08
DPWavLM [47]	82.18	53.65	12.48	6.61	5.58	10.66	29.69	34.84	5.56	7.18	36.53	42.88
[**Ours**] RD (DPWavLM)	**84.01**	**67.69**	**49.63**	**35.06**	**5.05**	**8.14**	**9.89**	**13.99**	**5.08**	**6.37**	**6.64**	**8.45**

LightHuBERT). Lastly, for the QbE experiments, the proposed methodology outperformed all the baseline distillation-only schemes and Teacher models, thus further corroborating the importance of the proposed method.

Robustness of Speaker-related Tasks

Next, Table 3.2 presents the results for the three speaker-related tasks. For these tasks, reverberation was shown to be a critical nuance factor for the downstream tasks, including the Teacher models, corroborating previous findings [109]. For instance, the SID task is based on the VoxCeleb1 [110] dataset, a classification problem with more than one thousand classes. Small distribution shifts can substantially affect system performance. Reverberation is known to affect, for example, pitch information, which could be conveyed by the different representations. Of the benchmark compression methods, LightHuBERT and DP-HuBERT-based methods again came out on top for non-robust Students, especially for the conditions involving reverberation. Notwithstanding, the proposed compression recipe achieved the best accuracy in these conditions, often outperforming the respective Teachers.

Table 3.3: Experimental results for semantics-related tasks on the SUPERB. Bold represents the best result and underline the second best for the compressed models.

Upstream	**IC** - ACC (↑)				**SF** - F1 (↑)				**SF** - CER (↓)			
	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)
Teachers												
Wav2Vec 2.0 Base [27]	92.83	69.26	68.78	50.88	88.30	74.06	70.20	47.61	24.77	41.73	48.47	68.11
HuBERT Base [100]	98.34	78.70	75.77	59.08	88.52	76.90	77.16	57.37	25.20	38.98	40.92	59.58
WavLM Base+ [28]	98.84	83.13	89.35	68.20	90.14	83.74	85.62	67.73	21.92	30.08	29.71	49.06
Robust HuBERT [50]	98.66	91.43	85.32	71.10	88.83	84.14	78.75	61.75	24.24	31.25	38.61	55.69
Distillation-only recipes												
Distiller (Wav2Vec 2.0) [45]	86.05	39.81	29.24	17.37	82.28	60.58	25.22	16.13	35.88	59.88	88.42	93.66
Distiller (HuBERT) [45]	94.12	48.72	47.72	24.31	82.12	64.14	34.43	20.83	35.21	55.44	82.26	90.56
Distiller (WavLM) [45]	94.89	63.25	48.43	31.69	84.76	68.74	36.72	21.43	32.89	50.60	80.47	89.66
Distiller (Robust HuBERT) [45]	**95.86**	67.07	46.30	28.34	**85.19**	68.71	29.08	17.06	**30.98**	49.69	85.84	92.92
FitHuBERT [85]	94.89	63.56	67.20	45.48	84.75	70.68	59.90	39.04	31.03	46.97	59.73	76.47
[**Ours**] RD (Wav2Vec 2.0)	89.30	71.47	71.00	54.47	80.61	73.68	72.12	60.32	37.58	46.92	49.71	62.51
[**Ours**] RD (HuBERT)	93.78	78.96	78.01	66.28	82.26	75.04	74.45	61.97	35.70	45.24	46.65	59.96
[**Ours**] RD (WavLM)	95.23	**82.05**	**86.45**	**72.21**	82.46	**75.93**	**75.38**	**62.88**	35.88	43.94	**45.85**	**58.55**
[**Ours**] RD (Robust HuBERT)	95.81	81.12	84.16	64.59	83.27	75.81	74.89	62.44	34.42	**43.73**	46.72	58.82
Distillation-*plus*-Pruning recipes												
LightHuBERT [46]	**98.60**	83.73	88.19	70.45	87.70	74.82	79.78	60.67	26.04	42.24	38.57	58.33
DPHuBERT [47]	97.89	73.45	72.29	50.25	87.41	72.91	68.74	44.37	28.36	44.85	52.17	72.89
DPWavLM [47]	98.58	83.94	77.72	61.24	88.03	78.25	75.05	55.23	**25.93**	38.72	44.78	63.15
[**Ours**] RD (DPWavLM)	98.39	**92.33**	**92.67**	**83.18**	**88.21**	**82.69**	**84.03**	**72.69**	26.80	**34.84**	**34.67**	**47.72**

Robustness of Semantic-related Tasks

For the semantic-related evaluations, Table 3.3 shows the results for two tasks: IC and SF. Baseline compression highly affected system performance in noisy conditions, with distillation-plus-pruning recipes again showing improved accuracy across all tested conditions. The proposed RobustDistiller consistently outperformed the original Distiller and FitHuBERT recipes in several cases, achieving results in line with LightHuBERT, DPHuBERT, or even surpassing the Teacher model (e.g., in the IC task under the n+r condition).

Robustness of Paralinguistic-related Tasks

Finally, for the ER task, Table 3.4 summarizes our findings. Compression was shown to be fairly resilient to the noisy conditions and resulted in recognition accuracy in line with those obtained with the Teacher models. The RobustDistiller method achieved the highest accuracy of all tested distillation-only compression models and was the closest to the accuracy of the Teacher models, despite requiring four times fewer parameters. FitHuBERT showed the best accuracy on the noise-only scenario among the compressed models.

Table 3.4: Experimental results for the paralinguistic task (emotion recognition) on the SUPERB. Bold represents the best result and underline the second best for the compressed models.

Upstream	**ER** - ACC ↑			
	(c)	(n)	(r)	(n + r)
Teachers				
Wav2Vec 2.0 Base [27]	63.01	51.46	32.70	27.02
HuBERT Base [100]	64.75	53.45	40.72	29.12
WavLM Base+ [28]	67.96	55.08	36.57	27.14
Robust HuBERT [50]	64.59	57.17	41.44	33.36
Distillation-only recipes				
Distiller (Wav2Vec 2.0) [45]	61.01	49.94	24.14	23.69
Distiller (HuBERT) [45]	62.55	47.89	31.35	25.77
Distiller (WavLM) [45]	62.09	51.14	29.24	25.27
Distiller (Robust HuBERT) [45]	**64.02**	51.60	29.29	25.08
FitHuBERT [85]	61.35	**52.17**	36.25	28.06
[**Ours**] RD (Wav2Vec 2.0)	63.57	50.50	**40.08**	29.98
[**Ours**] RD (HuBERT)	61.46	48.06	35.40	27.28
[**Ours**] RD (WavLM)	62.76	51.50	39.73	**32.65**
[**Ours**] RD (Robust HuBERT)	62.81	52.10	34.72	27.36
Distillation-*plus*-Pruning recipes				
LightHuBERT [46]	63.88	48.72	35.51	29.84
DPHuBERT [47]	63.06	52.90	35.45	27.44
DPWavLM [47]	**65.61**	57.58	36.87	29.96
[**Ours**] RD (DPWavLM)	64.87	**61.10**	**59.78**	**49.04**

Generalizing to other Teacher-Student recipes

The last subgroup in Tables 3.1 to 3.4 describes the results obtained from models that use distillation-plus-pruning recipes. Empirically, these models achieve the best overall performance among the compressed models, especially on content-related tasks (e.g., ASR). Beyond a similar generalization to its Teacher on clean scenarios, these models also exhibit some degree of robustness to adverse conditions. For instance, the LightHuBERT model is one of the most resilient to the reverberation condition, sometimes even outperforming its Teacher, the HuBERT model. In this work, to show how flexible our recipe is, we selected the DPWavLM model due to its excellent generalization capabilities and feasibility for training under academic resources. DPWavLM consists of two stages. First, distillation and pruning are jointly performed. In the second stage, only distillation is used to

adjust the Student. We specifically insert our two modifications of the RobustDistiller in the second stage.

Our results show that the RobustDistiller recipe can increase the robustness of the Student model and help with the generalization for clean scenarios, as observed in several tasks such as ASR, KS, SID, and SD. For the noisy scenarios, the gap between clean and noisy settings is significantly smaller. For instance, the RobustDistiller applied to the DPWavLM framework in the noise-plus-reverberation scenario for the IC task has a drop of 15.21% percentage points in accuracy. In contrast, Robust HuBERT, a robust model four times the size of our Student, has a drop of 27.56%.

Overall Performance across tasks and scenarios

Lastly, Table 3.5 shows the overall score (Eq. 1) grouped by task type and evaluation scenario. We also present the number of parameters and the number of multiply-accumulate operations (MACs) for each model. As expected for models tested under clean settings, the Teacher models have better overall performance when compared to their smaller counterparts. For the compressed models, we observe that distillation-plus-pruning takes the lead across clean tasks, with a commendable performance for DPWavLM and LightHuBERT. Nonetheless, the proposed RobustDistiller shows that it can not only improve "in-the-wild" scenarios, but also help the model generalize better for clean settings. For instance, for content- and speaker-related tasks, DPWavLM trained with the proposed recipe improves the overall performance under clean scenarios, with 17.15% and 20.22% increase in the score, respectively. Speaker-related tasks, in general, greatly benefit from the proposed robust distillation recipe for both distillation-plus-pruning and distillation-only methods.

For noisy conditions, we observe that RobustDistiller can drastically improve across all four groups of tasks, sometimes even surpassing robust Teacher models, such as WavLM and Robust HuBERT. Comparing non-robust distillation-only methods with the proposed RobustDistiller on semantics-related tasks with reverberation, we can observe a Score gain of 2.6 times. Similar results can be observed on speaker-based tasks. Overall, the gains achieved with RobustDistiller motivate the development of distillation methods with a robustness-first principle during its development.

Table 3.5: Overall Score results for each scenario on the original 10 SUPERB tasks grouped by category. Bold represents the best result and underline the second best for the compressed models.

Upstream	#params	MACs	Content-related Tasks				Speaker-related Tasks				Semantic-related Tasks				Paralinguistics-related Tasks			
	(M)	(G)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)
Teachers																		
Wav2Vec 2.0 Base [27]	95	1669	841	686	750	712	459	221	260	214	671	576	668	533	289	270	240	131
HuBERT Base [100]	95	1669	702	630	605	600	605	450	431	398	885	719	784	685	539	421	465	214
WavLM Base+ [28]	95	1670	814	780	817	759	868	752	560	466	1000	908	974	857	1000	545	349	136
Robust HuBERT [50]	95	1669	604	827	726	697	651	678	431	405	921	982	876	817	516	703	485	382
Distillation-only recipes																		
Distiller (Wav2Vec 2.0) [45]	23	786	123	0	37	123	248	143	0	0	71	0	0	0	0	155	0	0
Distiller (HuBERT) [45]	23	786	289	314	242	254	371	338	254	247	393	160	210	90	222	0	202	82
Distiller (WavLM) [45]	23	786	362	461	290	224	579	515	131	123	529	388	233	154	156	246	143	62
Distiller (Robust HuBERT) [45]	23	786	**405**	405	132	121	471	483	148	132	**609**	431	161	91	**433**	281	145	55
FitHuBERT [85]	22	3122	395	455	365	331	67	126	257	219	559	442	565	408	50	324	340	172
[**Ours**] RD (Wav2Vec 2.0)	23	786	195	628	740	695	413	702	885	898	127	549	688	647	368	198	**447**	248
[**Ours**] RD (HuBERT)	23	786	265	**781**	**851**	**871**	555	819	**928**	**942**	375	649	766	758	65	13	316	141
[**Ours**] RD (WavLM)	23	786	346	700	795	833	**633**	**830**	915	921	434	**699**	**840**	**814**	253	273	437	**354**
[**Ours**] RD (Robust HuBERT)	23	786	363	742	839	785	499	792	908	909	502	690	816	753	259	**319**	297	145
Distillation-*plus*-Pruning recipes																		
LightHuBERT [46]	27	861	538	585	754	652	473	345	634	546	861	717	903	792	413	63	319	243
DPHuBERT [47]	24	654	459	387	451	489	516	380	420	355	788	577	674	488	295	379	317	148
DPWavLM [47]	24	589	462	524	470	512	638	614	321	266	**870**	785	774	672	**662**	734	357	247
[**Ours**] RD (DPWavLM)	24	589	**541**	**792**	**845**	**852**	**766**	**917**	**1000**	**1000**	854	**945**	**972**	**1000**	556	**1000**	**1000**	**1000**

3.5.2 Performance on SUPERB-SG Tasks: Enhancement and Source Separation

For downstream generative tasks, our assumption is that feature denoising KD and multi-task learning will lead to improved disentanglement in the latent space, resulting in improved results. Table 3.6 reports the results obtained for the SE and SS tasks. One of the first observations for these tasks is that compressed models do not have a significant performance gap compared to their Teacher model, as has happened on other tasks. For distillation-only recipes, the proposed modifications led to improvements in both SE and SS compared to their original recipe. For distillation-plus-pruning recipes, however, there is no clear pattern on whether RobustDistiller helps or not. Furthermore, examining the overall results for smaller models, robust or not, these experiments suggest that there is still room for improvements on generative tasks.

3.5.3 Speech-noise disentanglement

The effects of feature denoising knowledge distillation and the multi-task learning during distillation proved especially useful in clean and noisy settings. Initially, we hypothesized that both paradigms helped the model to better generalize by reinforcing the disentanglement of speech features from noise factors. In Figure 3.2, we further explore this hypothesis and test the effects that noise has on the latent space of three different models: HuBERT, Distiller (HuBERT), and RobustDistiller (HuBERT), from top to bottom, respectively. Left-side plots are for the clean utterances and

Table 3.6: Experimental results on the SUPERB-SG benchmark for Speech Enhancement (SE) and Source Separation (SS) tasks. Bold represents the best result and underline the second best for the compressed models.

Upstream	SE			SS	Score (↑)
	STOI (↑)	PESQ (↑)	SI-SDR (↑)	SI-SDR (↑)	
Teachers					
Wav2Vec 2.0 Base [27]	94.83	2.93	9.34	10.02	715.07
HuBERT Base [100]	94.86	2.98	9.05	9.97	726.45
WavLM Base+ [28]	95.17	3.04	9.41	11.37	988.09
Robust HuBERT [50]	94.93	3.00	9.16	9.86	751.27
Distillation-only recipes					
Distiller (Wav2Vec 2.0) [45]	94.71	2.91	9.41	9.48	633.51
Distiller (HuBERT) [45]	94.76	2.96	9.24	9.39	656.65
Distiller (WavLM) [45]	94.88	2.98	8.71	10.03	702.04
Distiller (Robust HuBERT) [45]	94.85	2.94	9.44	9.32	671.14
FitHuBERT [85]	94.31	2.79	7.75	6.02	0.00
[**Ours**] RD (Wav2Vec 2.0)	94.89	2.99	9.19	10.01	752.93
[**Ours**] RD (HuBERT)	94.90	**3.00**	**9.47**	9.80	770.31
[**Ours**] RD (WavLM)	**94.95**	2.99	9.46	**10.17**	**802.38**
[**Ours**] RD (Robust HuBERT)	94.82	2.99	9.14	9.73	711.64
Distillation-*plus*-Pruning recipes					
LightHuBERT [46]	94.72	2.96	9.11	9.91	683.06
DPHuBERT [47]	94.81	2.92	9.09	9.80	662.85
DPWavLM [47]	94.92	**3.02**	9.52	**10.56**	**861.84**
[**Ours**] RD (DPWavLM)	**94.95**	3.00	**9.53**	10.47	846.80

right-side for noise under a 0 dB SNR setting. Each individual point is an utterance, and its shape/color represents one out of eight randomly-selected speakers from the LibriSpeech dataset. First, we compute a global average pooling of each frame of the tested representation. Then we reduce the dimensionality of the signal to two dimensions via a tSNE transformation. As can be seen, the representations are able to accurately discriminate the speakers in the clean condition. In turn, under the noisy setting, only the RobustDistiller method preserves the disentanglement across speakers.

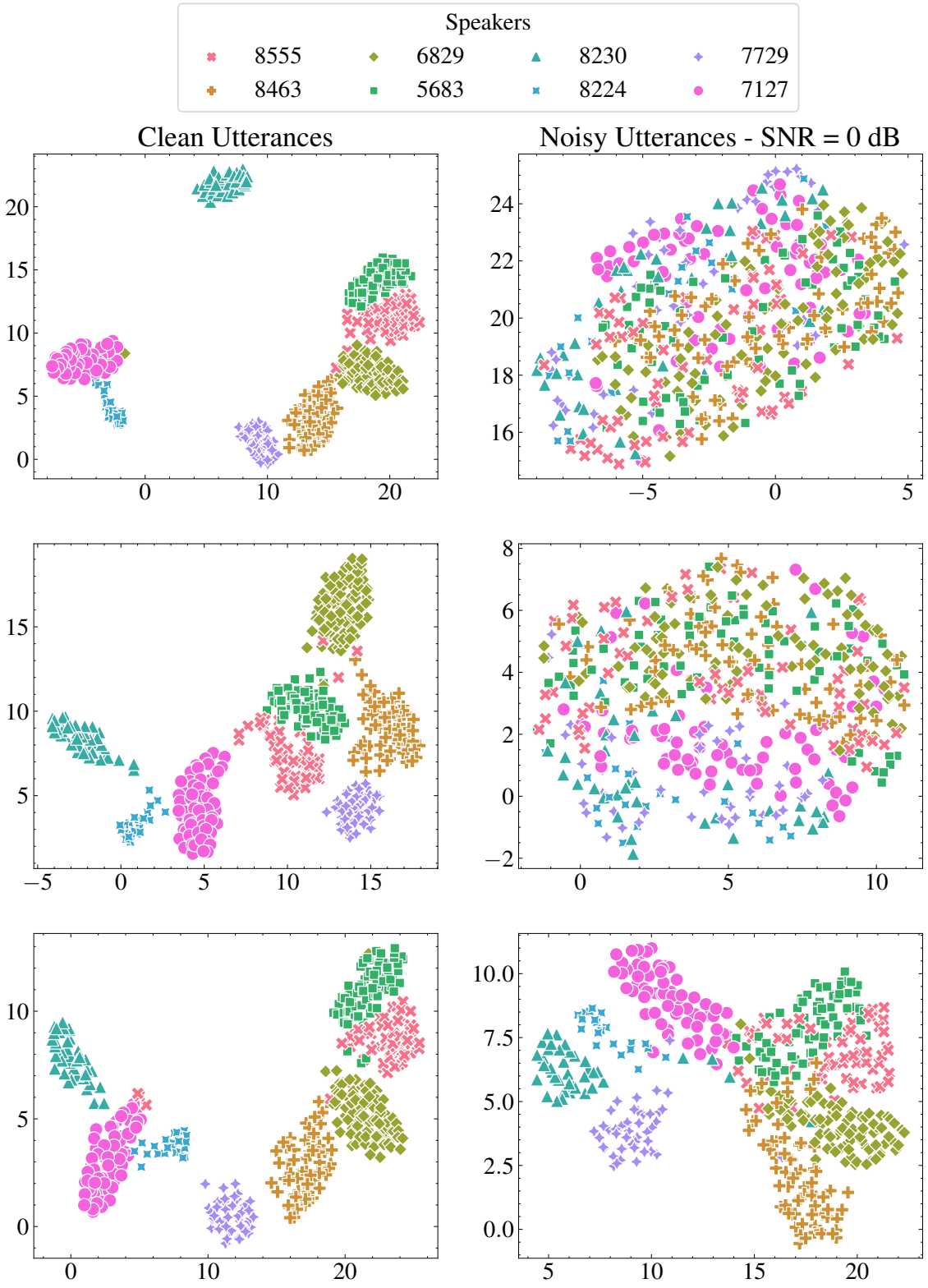


Figure 3.2: tSNE visualization on the effect of noise on the latent representation of (top to bottom) HuBERT, Distiller (HuBERT), and RobustDistiller (HuBERT), respectively. Clean signals are shown on the left, while noisy on the right. Data are from eight randomly-selected speakers of LibriSpeech.

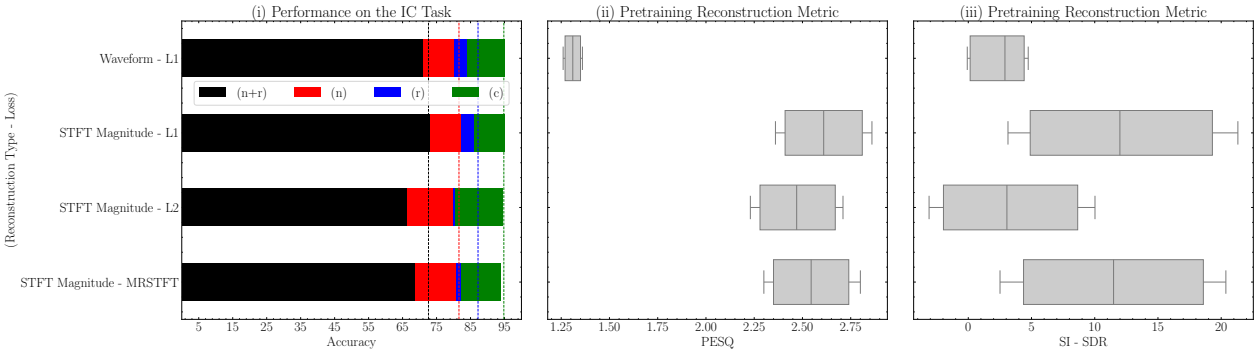


Figure 3.3: Measuring the impact of different enhancement heads and loss functions for our proposed multi-task learning paradigm.

3.5.4 Impact of the Enhancement Head

To evaluate the effectiveness of the proposed modification #2 (i.e., adding the enhancement head), we conducted experiments with models capable of reconstructing either the original waveform or the clean STFT magnitude, using different enhancement losses. During the distillation phase, we measured reconstruction metrics (e.g., PESQ and SI-SDR) every 50 steps for each model configuration. Once the distilled model was obtained, we fine-tuned it for three different tasks: KS, IC, and ER.

Figure 3.3 (i) depicts the results for the IC task under the four tested scenarios. The dashed lines indicate the outcome for a model without the enhancement head but which has the feature denoising knowledge distillation step. Figure 3.3 (ii) and (iii) display the pretraining PESQ and SI-SDR, respectively.

As can be seen, the combination that achieves the best overall performance relies on an enhancement head that estimates the binary mask for the STFT magnitude and employs an L1-loss function. This result highly correlates with the best reconstruction metrics observed during the pretraining phase. We hypothesize that, for small-capacity enhancement heads that estimate a binary mask, the greater the reconstruction during the training, the better the final disentanglement because the Student model is learning to preserve more speech features in their latent space. In particular, for the IC task in Figure 3.3(i), we observe an improvement in three of the four tested scenarios using the enhancement head. For the reverberation scenario, the usage of our proposed enhancement head slightly improved upon the setting where we use only feature denoising KD; thus, a limitation of this work is to experiment with better architectures that can handle better dereverberation.

Despite achieving poor reconstruction metrics during pretraining, the second-best overall result across the three tasks goes for the model reconstructing the raw waveform directly and using an L1-loss function. Going from a very compressed latent representation to a high-dimensional signal (i.e., raw waveform) is difficult. Furthermore, inserting a more robust enhancement head can soften the job from the Student model, thus undermining the full potential of creating an encoder that disentangles noise from speech features as we planned.

3.5.5 Discussion on distilling from already robust Teachers

One important question that may arise from the analysis herein is the following: What if we distill a representation that was built to be robust; do we get better results than developing a robust distillation process? To answer this question, we zoom into the results in Table 3.5. In this analysis, we consider the WavLM and Robust HuBERT models to be robust Teachers since both were augmented with noise during the training phase. First, considering distillation-only recipes, we observe that the Distiller recipe with HuBERT and Robust HuBERT leads to the former Student being more robust in the noise-only setting. However, both have a huge gap compared to the Teacher model. When applying our methodology, RobustDistiller, we observe a large improvement in all three noisy scenarios.

Interestingly, RobustDistiller with HuBERT or the Robust HuBERT models generate Students with similar performance. Thus, a two-step process of first making the Teacher model and then distilling it is not necessary. It is computationally better to distill a non-robust model directly with RobustDistiller, and we can achieve similar results.

Lastly, similar results can be observed when using the distillation-plus-pruning methods. From the beginning, DPWavLM can preserve much of the robustness of WavLM. However, if trained with RobustDistiller, we can further improve its performance in noisy settings and also its generalizability in clean scenarios, as observed for content- and speaker-related tasks.

3.5.6 Effect of noise type in low SNR regimes

Results above have shown the sensitivity of different upstream models to the presence of noise during testing. An in-depth analysis showed that this effect was particularly true at SNR regimes below

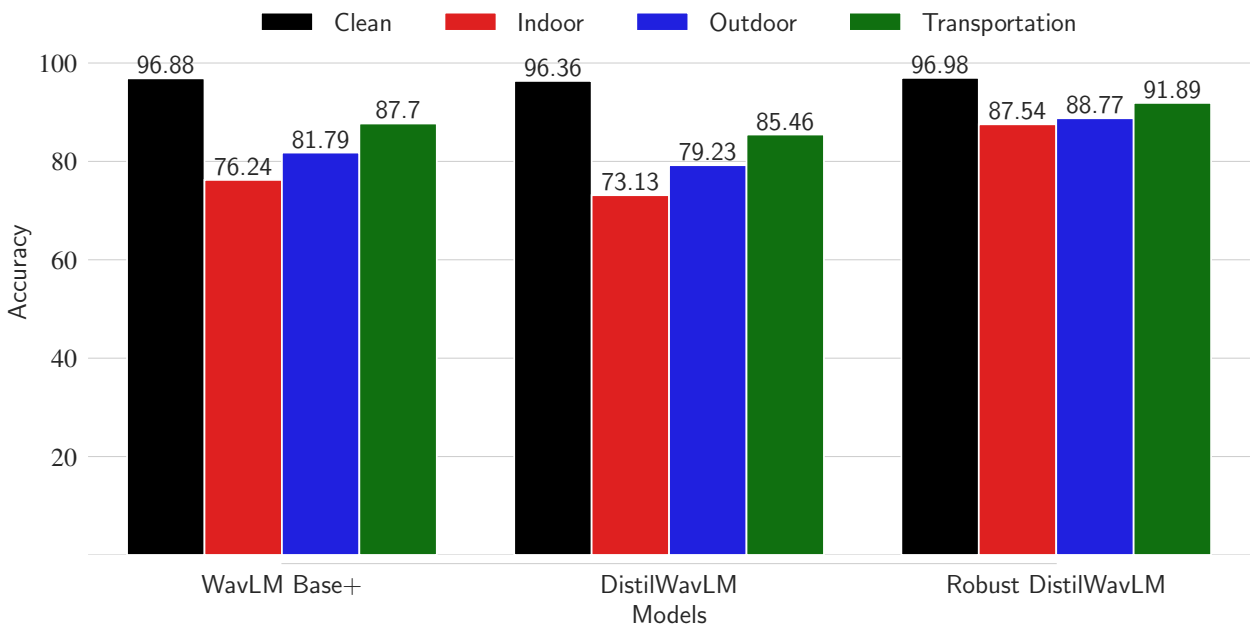


Figure 3.4: KS accuracy per noise type with clean speech levels used for referencing.

an SNR of 10 dB. Here, we explore if the noise type under these SNR regimes plays a role in the final system accuracy. For simplicity, here we focus only on the KS task and noise levels at the SNR range of $[-5, 5]$ dB. The DCASE2020 noise dataset used for testing was comprised of noise signals under three categories: indoor (e.g., shopping mall, airport), outdoor (e.g., park, public square), and transportation (e.g., bus, metro, tram). Figure 3.4 shows the accuracy achieved under the clean condition, as well as the three different noise types for the WavLM Base+ Teacher model, as well as this model distilled with the benchmark distiller and the proposed RobustDistiller frameworks. As can be seen, for the benchmark systems, indoor noise causes the most significant decline in accuracy, which could be attributed to the combined impact of reverberation and additive noise. Transportation noise had the least impact on the benchmark methods. The proposed approach, in turn, resulted not only in the highest overall accuracy across the three noise types, but showed to be independent of noise type, thus further validating the noise robustness properties of the proposed solution.

3.5.7 Effects of room size

Lastly, we explore the impact of room size on system performance. Room sizes are directly proportional to reverberation times, with larger rooms typically resulting in lower performance. For

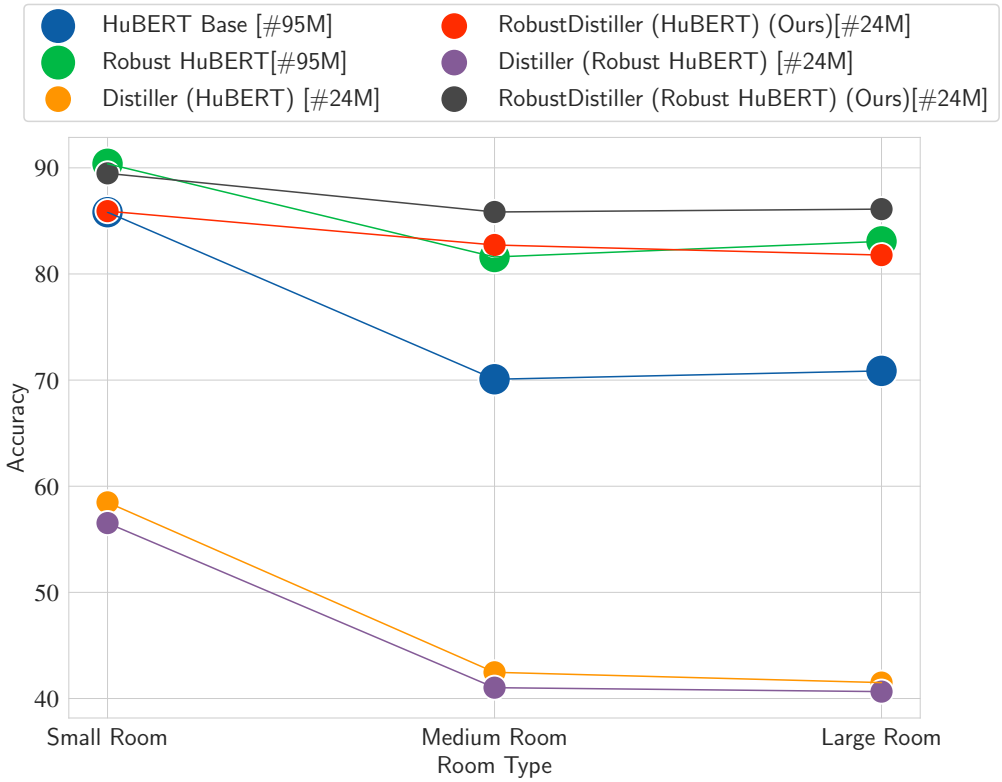


Figure 3.5: IC performance per room size for different HuBERT distillation schemes.

this investigation, we use the IC task and HuBERT-based variations and the recorded RIRs from small, medium, and large-sized rooms, as described in Section 3.4.1. Figure 3.5 depicts the accuracy achieved for each of the three room sizes. As expected, accuracy decreases as room sizes increase. The two proposed RobustDistiller setups achieved the highest accuracy across the three room sizes, with accuracy values at times outperforming the Teacher model, whilst requiring just one-quarter of the parameters. These findings are promising and suggesting that robust distillation for edge speech applications may indeed be possible via a robust distillation process.

3.5.8 Ablation on Curriculum Learning experiments

In this section, we analyze the impact of the progressive learning strategy originally proposed in [97]. It is important to note that, due to methodological refinements in the current work [48] (e.g., expanding the SNR range during training), the absolute performance metrics reported here may differ slightly from subsection 3.5.1; however, the overarching trends remain consistent.

Table 3.7: Curriculum Learning: Experimental results for keyword spotting, intent classification, and emotion recognition under clean (c), noisy (n), reverberation (r), and noise-plus-reveberation (n+r) test conditions.

		Keyword Spotting				Intent Classification				Emotion Recognition			
Upstream	#params (M)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)	(c)	(n)	(r)	(n + r)
HuBERT Base	95	96.30	84.10	61.70	53.49	98.34	80.78	75.77	56.60	64.75	52.57	40.72	32.06
Robust HuBERT	95	96.33	91.46	74.81	66.80	98.66	92.75	85.32	71.53	64.59	57.83	41.44	34.83
Distiller (HuBERT) [45]	24	96.14	83.45	59.20	52.26	94.12	54.23	47.72	28.37	62.55	48.67	31.35	28.30
[**Ours**] RD (HuBERT)													
W/O Curriculum-Learning	24	96.59	92.31	85.98	79.52	94.15	81.68	83.42	67.23	61.71	50.89	40.83	33.95
W/ Curriculum-Learning	24	96.92	91.85	85.30	79.00	93.91	83.39	81.18	64.70	62.96	51.48	39.24	32.78

Table 3.7 presents the ablation results for RobustDistiller with and without the Curriculum Learning (CL) strategy. The results indicate that CL offers selective benefits, primarily improving performance in clean and noise-only conditions for Intent Classification and Emotion Recognition. For example, in the Intent Classification task, the CL-trained model achieves an accuracy of 83.39% under noisy conditions, outperforming the non-curriculum baseline of 81.68%.

However, these benefits do not extend to reverberant environments. Across all tasks, we observe a consistent performance degradation in the reverberation (**r**) and noise-plus-reverberation (**n+r**) scenarios when CL is employed. For instance, in Keyword Spotting, performance under reverberation drops from 85.98% (w/o CL) to 85.30% (w/ CL). We hypothesize that this discrepancy stems from the varying degrees of control over the augmentation parameters. While the curriculum effectively regulates the Signal-to-Noise Ratio (SNR), gradually introducing more challenging noise levels, the reverberation time (RT60) is sampled stochastically without an explicit difficulty progression (we do not have RT60 measurements for the RIR samples). Consequently, the model may fail to learn robust de-reverberation features as effectively as it learns denoising features.

Finally, from a computational perspective, we empirically observe that Curriculum Learning accelerates convergence, allowing the model to reach lower training loss values in fewer iterations. However, given sufficient training time, both strategies ultimately converge to comparable final loss minima, suggesting that CL primarily acts as an efficiency optimization rather than a strict performance upper-bound lifter in this context.

3.6 Conclusion

In this Chapter, we described our work, RobustDistiller, a methodology to improve the environmental robustness of the distillation process of speech representation learning algorithms. In particular, two modifications are proposed to the original distillation recipes: (i) Feature Denoising Knowledge Distillation; and (ii) multi-task learning, where a speech enhancement step is jointly performed to reinforce the disentanglement of speech features from noise factors. We show that our recipe can be adapted to both layerwise distillation (e.g., DistilHuBERT) and distillation-plus-pruning (e.g., DPHuBERT) methods. We test the robustness of our model across clean and noisy conditions on top of 12 speech-processing tasks from SUPERB and SUPERB-SG. Experimental results show the proposed method frequently outperforming the original Teacher models with roughly four times fewer parameters. Overall, the proposed RobustDistiller on top of the DPWavLM recipe resulted in the best performance across tasks, thus suggesting that their use could be suitable for "in-the-wild" edge speech applications.

Chapter 4

Adversarially robust features via canonical correlation analysis

4.1 Preamble

This chapter is compiled from material extracted from the following publications: Our conference paper at the 2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC) [57], and the conference paper at the 2024 IEEE International Conference on Acoustics, Speech, & Signal Processing (ICASSP) [111].

4.2 Introduction

Keyword spotting (KS) systems are usually the first layer of virtual assistants, where such models attempt to identify preset words in an utterance to activate the device (e.g., "Hey Siri"). On-device voice-controlled assistants, such as Google Now and Alexa, are becoming ubiquitous in our daily lives; however, as discussed earlier, they can pose challenges to energy efficiency, real-time processing, and user privacy. For KS, the use of self-supervised speech encoders has become a popular tool. Speech encoders, such as *Wav2Vec 2.0* [27] and WavLM [28], are designed to learn combined acoustic and language models from continuous audio inputs.

However, the deployment of these systems faces a critical security bottleneck: vulnerability to adversarial attacks. Recent research indicates that KS systems built upon SSL backbones are highly susceptible to malicious perturbations during inference [57]. Crucially, these threats extend to the physical domain; over-the-air attacks have demonstrated that imperceptible noise added to a user's voice in the physical environment can successfully trigger misclassifications [112]. While defense mechanisms like Adversarial Training (AT) [52] and TRADES [113] exist (as detailed in Chapter 2), applying them to edge devices remains an open issue.

Designing models that are simultaneously compact, accurate, and robust is a challenging open problem. As observed in [52, 114], adversarial robustness is often at odds with model capacity. While it is possible to train small models directly with AT, this typically results in suboptimal performance on clean data (the robustness-accuracy trade-off). Robust knowledge distillation methods, such as ARD [54] and RSLAD [55], attempt to mitigate this by leveraging a strong Teacher for clean settings. However, neither method consistently outperforms a simple small model trained with AT, suggesting that current distillation paradigms fail to effectively transfer robustness to ultra-low-resource Students.

To address this limitation, we propose VIC-KD, a novel distillation framework that compresses speech encoders while significantly enhancing their adversarial robustness against digital-based attacks. Physical attacks (e.g., over-the-air attacks) are not considered in this study and remain an interesting direction for future work. In particular, we demonstrate that incorporating multi-view inputs and imposing geometric constraints on the Student's latent space are essential for bridging the robustness gap. We focus on an extreme compression regime, targeting Student models with fewer than 96K parameters and 3 million MACs. Distilling from fine-tuned versions of *Wav2Vec 2.0* and WavLM, our results show that VIC-KD achieves superior robustness compared to traditional defenses like TRADES [113] and outperforms state-of-the-art robust distillation methods (ARD and RSLAD).

4.3 Variance-Invariance-Covariance Knowledge Distillation

Herein, instead of relying solely on the Teacher model to guide how the Student logits should behave, we induce geometric properties of the Student's latent space via variance-invariance-covariance

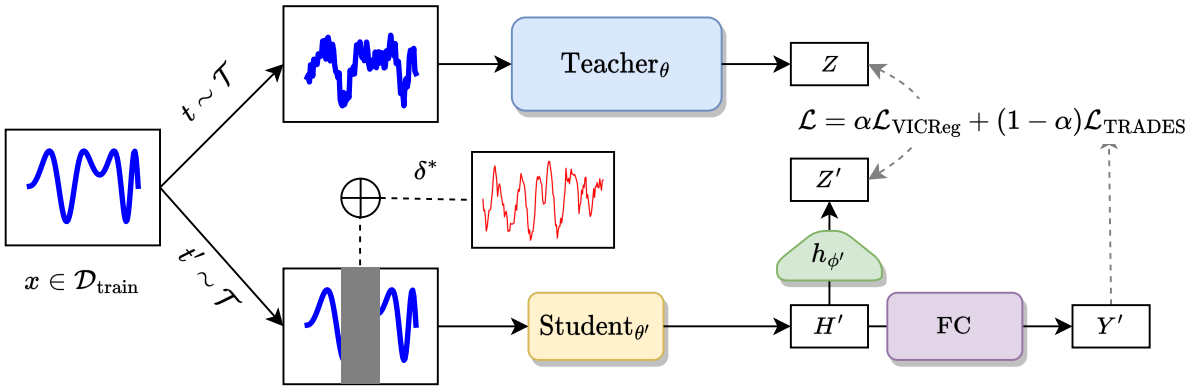


Figure 4.1: Diagram of the proposed robust distillation recipe: VIC-KD.

regularization (VICReg) [58] and the use of multi-view inputs to each model. Although the use of multi-view inputs for KD has already been explored in the literature to increase the environmental robustness of Student models [90, 48], its effect on robust distillation remains to be determined. Figure 4.1 depicts a block diagram of the proposed VIC-KD distillation recipe.

In the SSL literature, joint embedding architectures (JEA) are becoming popular due to their effectiveness in learning latent factors from data [115]. VICReg is one such method that preserves the information in embeddings while avoiding representational collapse. Here, we expand those ideas to a robust distillation method. First, we sample an utterance from $x \sim \mathcal{D}_{\text{train}}$ and two random transformations $\{t, t' \sim \mathcal{T} | t \neq t'\}$. As depicted in the upper branch of Figure 4.1, the Teacher model T_θ is responsible for extracting representations from the speech input. More precisely, representation Z is extracted, which is the weighted sum of the intermediate representations from all Transformer layers, and aggregated over the time dimension.

The Student branch, on the other hand, receives the other utterance view generated by t' with an added adversary perturbation generated via a 10-step Projected Gradient Descent (PGD) [52]. First, for both inputs, we extract the latent representations of the Student model, denoted by H'. This representation is then fed to a classification head, generating the model logits Y' that attempt to predict the spoken commands. In parallel, H' is fed to a projection head responsible for generating Z', which has the same dimensionality as the Teacher latent representation. Note that, at test time, only the Student encoder $S_{\theta'}$ and the classification head are used.

In addition to the mechanism described above, we now describe the contribution of each term that composes the three-factor VIC-KD loss function, presented in Equation 4.1: (1) variance, (2)

Table 4.1: Baseline results obtained directly in a supervised setting without knowledge distillation.

Model	Label	#Params	MACs
Wav2Vec 2.0 - Base	T1	94.4M	6.2G
WavLM - Base+	T2	94.4M	6.1G
TC-ResNet	S1	67.7K	1.6M
XVector	S2	80.4K	2.9M

invariance, and (3) covariance terms. Following our previous notation, we define $Z = [z_1, z_2, ..., z_n]$ and $Z' = [z'_1, z'_2, ..., z'_n]$ as the d-dimensional latent representations generated by the Teacher and Student models for natural and adversarially perturbed input, respectively, for a batch of n utterances. First, the invariance term consists of the mean-squared error between the two latent representations, Z and Z'. Next, the variance term induces the learned latent embedding to vary within the other batch elements, thus avoiding collapse on the same vector. Lastly, the covariance term prevents the Student model, which already has limited capacity, from encoding similar features. Mathematically, this can be represented as:

$$\mathcal{L}_{\text{VICReg}} = \text{Var}(Z') + \text{Inv}(Z, Z') + \text{Cov}(Z'), \tag{4.1}$$

$$\text{Var}(Z') = \frac{1}{d}\sum_k \max\left(0, 1 - \sqrt{\text{Var}(z'_k) + \epsilon}\right), \tag{4.2}$$

$$\text{Inv}(Z, Z') = \frac{1}{n}\sum_k ||z_k - z'_k||_2^2, \tag{4.3}$$

$$\text{Cov}(Z') = \frac{1}{d}\sum_{i \neq j}\left[\frac{1}{n-1}\sum_{k=1}^{n}(z'_k - \bar{z}')(z'_k - \bar{z}')^\intercal\right]_{i,j}^2. \tag{4.4}$$

In our experiments, we consider an equal contribution of each term in the final VICReg loss. The interested reader is referred to [58] for more details about the loss implementation. Finally, the VIC-KD loss is computed as the convex combination between the TRADES [113] and VICReg losses, controlled through a hyperparameter α as described in Figure 4.1.

4.4 Experimental Setup

To train the KS system, we use the *Google Speech Commands v0.02* (GSC) dataset [59]. GSC is a widely used dataset in the literature and comprises about 100k 1-second utterances across 35 commands, sampled at 16 kHz. We consider two versions of the dataset, one with all commands

Table 4.2: Baseline results for the Teacher models in a supervised setting without knowledge distillation.

	Natural Training				Robust Training			
	GSC v12		GSC v35		GSC v12		GSC v35	
Model	Clean	AutoAttack	Clean	AutoAttack	Clean	AutoAttack	Clean	AutoAttack
T1	99.07	9.40	97.53	2.68	97.58	93.80	95.77	90.63
T2	98.94	6.48	97.42	0.64	97.95	94.79	97.20	93.26

Table 4.3: Baseline results for the Student models in a supervised setting without knowledge distillation.

	Natural Training					Robust Training				
Model	Clean	FGSM	PGD-30	AutoAttack	FAKEBOB	Clean	FGSM	PGD-30	AutoAttack	FAKEBOB
					GSC v12					
S1	94.95	61.78	2.67	2.67	39.54	93.69	88.37	80.77	79.25	68.90
S2	96.43	67.97	3.48	3.09	29.75	94.70	89.45	87.13	80.33	62.46
					GSC v35					
S1	93.53	63.11	0.06	0.09	35.77	92.95	86.26	77.94	77.30	62.38
S2	94.54	69.26	0.67	1.36	21.92	93.56	87.19	83.64	78.27	54.11

and another with only 12 classes that include labels such as {yes, no, up, down, left, right, on, off, stop, go, unknown, and silence}. The *silence* class is made up of background noises with no speech, while the *unknown* label includes utterances uniformly sampled from unused classes. We follow the SpeechBrain [60] recipe, splitting the dataset into train, validation, and test sets at 80%, 10%, and 10%, respectively.

In this study, we examine two Teacher models, *Wav2Vec 2.0* and WavLM, along with two Student models, TC-ResNet-8 [61] and a custom XVector [62], with reduced numbers of time-delay neural (TDNN) layers and smaller kernel sizes to meet our resource constraints. Furthermore, as summarized in Tables 4.2 and 4.3, we present results for conventional training and AT via TRADES. Moreover, we report accuracy for clean speech files, as well as speech files corrupted by the AutoAttack [116], an ensemble method comprised of APGD, APGD-T, and FAB adversarial attacks [116, 117]. Later experiments in this chapter rely on these reference values for comparison.

Baseline models are trained for 100 epochs, with a batch size of 32 samples, using an Adam optimizer, and learning of 10^{-3} that linearly decays to 10^{-4}. Conversely, we fine-tune the Teacher model for the KS task for 10 epochs, and the learning rate is scheduled from 5×10^{-4} to 5×10^{-5}. The Teacher model consists of the S3RL encoder with a weighted sum and aggregation mechanism described previously and a simple linear layer that classifies the input speech signal into the desired

Table 4.4: Results on the GSC dataset with 12 classes for distillation methods. Distilling from standard and robust Teachers.

Teacher / Student	Standard Teacher					Robust Teacher				
	Clean	FGSM	PGD_{30}	AutoAttack	FAKEBOB	Clean	FGSM	PGD_{30}	AutoAttack	FAKEBOB
Knowledge Distillation (KD) [42]										
T1 / S1	95.80	81.38	8.94	5.54	47.43	95.61	80.15	5.38	4.49	43.34
T1 / S2	96.77	84.26	11.09	7.59	52.28	96.68	82.62	9.18	6.84	46.78
T2 / S1	96.20	80.81	6.16	9.29	45.89	95.70	81.16	5.63	5.11	44.86
T2 / S2	96.91	84.03	11.73	8.00	49.77	96.62	82.89	7.90	7.19	44.29
Adversarial Robust Distillation (ARD) [54]										
T1 / S1	94.54	89.42 (+1.1%)	75.90 (-4.9%)	76.41 (-2.8%)	72.08 (+3.2%)	94.92	89.58 (+1.2%)	73.52 (-7.3%)	74.16 (-5.1%)	69.34 (+0.4%)
T1 / S2	95.37	91.60 (+2.1%)	81.92 (-5.2%)	79.74 (-0.6%)	71.87 (+9.4%)	95.71	90.67 (+1.2%)	83.20 (-3.9%)	79.94 (-0.4%)	70.99 (+8.5%)
T2 / S1	95.17	89.82 (+1.4%)	77.47 (-3.3%)	74.13 (-5.1%)	73.42 (+4.5%)	95.30	90.47 (+2.1%)	76.93 (-3.8%)	76.87 (-2.4%)	73.11 (+4.2%)
T2 / S2	95.27	89.93 (+0.5%)	81.01 (-6.1%)	78.33 (-2.0%)	71.42 (+9.0%)	95.29	88.85 (-0.6%)	76.69 (-10.4%)	74.03 (-6.3%)	69.93 (+7.5%)
Robust Soft-Label Adversarial Distillation (RSLAD) [55]										
T1 / S1	94.65	90.08 (+1.7%)	77.99 (-2.8%)	78.06 (-1.2%)	71.55 (+2.6%)	94.59	88.86 (+0.5%)	76.70 (-4.1%)	76.27 (-3.0%)	69.81 (+0.9%)
T1 / S2	95.37	90.71 (+1.3%)	84.32 (-2.8%)	80.79 (+0.5%)	69.24 (+6.8%)	95.45	90.86 (+1.4%)	84.42 (-2.7%)	81.98 (+1.7%)	69.09 (+6.6%)
T2 / S1	95.11	88.36 (+0.0%)	74.12 (-6.6%)	77.17 (-2.1%)	68.45 (-0.5%)	94.11	89.23 (+0.9%)	78.14 (-2.6%)	77.99 (-1.3%)	72.14 (+3.2%)
T2 / S2	95.34	90.88 (+1.4%)	84.48 (-2.6%)	81.30 (+1.0%)	70.27 (+7.8%)	95.69	91.19 (+1.7%)	84.53 (-2.6%)	81.90 (+1.6%)	70.26 (+7.8%)
VIC-KD (**Ours**)										
T1 / S1	95.31	91.10 (+2.7%)	84.18 (+3.4%)	83.75 (+4.5%)	74.18 (+5.3%)	95.48	90.84 (+2.5%)	84.24 (+3.5%)	83.38 (+4.1%)	73.82 (+4.9%)
T1 / S2	**96.50**	92.90 (+3.5%)	87.79 (+0.7%)	**86.12 (+5.8%)**	**77.13 (+14.7%)**	96.29	**92.93 (+3.5%)**	87.59 (+0.5%)	85.33 (+5.0%)	75.82 (+13.4%)
T2 / S1	95.39	91.21 (+2.8%)	83.89 (+3.1%)	83.30 (+4.1%)	74.10 (+5.2%)	95.37	90.82 (+2.4%)	84.69 (+3.9%)	83.73 (+4.5%)	75.39 (+6.5%)
T2 / S2	95.92	**93.05 (+3.6%)**	**88.04 (+0.9%)**	86.08 (+5.8%)	74.87 (12.4%)	**96.39**	92.86 (+3.4%)	**88.17 (+1.0%)**	**86.31 (+6.0%)**	**76.92 (+14.5%)**

number of classes. Note that, for VIC-KD, we can discard the Teacher's linear layer since we are interested in the latent representation. For adversarial attacks, we rely on FGSM [118], PGD (30-steps) [52], FAKEBOB [119], and AutoAttack [116]. We consider ℓ_∞ bounded attacks with an $\epsilon = 1.5 \times 10^{-3}$.

Three distillation methods, namely KD, ARD, and RSLAD, are used to benchmark the proposed technique. Training is performed for 250 epochs with parameters matching the small baseline models, as specified above. VIC-KD relies on a 10-step PGD-like attack, also with an $\epsilon = 1.5 \times 10^{-3}$ and a step size of 3×10^{-4}, to generate the adversarial perturbation for training. Additionally, multi-view inputs are used for Student and Teacher models, incorporating clean data, noise, reverberation, noise-plus-reverberation, wave chunk dropping (i.e., randomly setting segments of the waveform to zero), and speed perturbation into the transformations $\mathcal{T}$.

4.5 Experimental Results and Discussion

Herein, we describe the obtained results related to our investigation on adversarial robustness induced by our methodology.

4.5.1 Result #1: Robust accuracy

Table 4.4 presents the main experimental results exploring two scenarios: standard vs. robust Teachers as guides. We report clean and robust accuracies for each robust distillation method, with differences relative to the respective baseline in parentheses. In fact, given the low robust accuracy and that KD is not a robust method, we do not compute its delta against the robust baseline. KD with standard Teacher outperforms baselines and other robust methods in clean accuracy. However, despite some improvement, the robust accuracy is still below random guess for all attacks, except for FGSM, which is a very simple one. ARD, on the other hand, significantly enhances overall robust accuracy, but still falls short of the baseline performance under the robust condition.

Regardless, we observe an improvement in the overall scenario for RSLAD and VIC-KD, which are built upon TRADES. For the Student model with bigger capacity, e.g., the XVector (S2), we observe the first results where the model can improve upon the baseline for robust accuracy. The TC-ResNet (S1) on the RSLAD distillation has inline results with the baseline, but improves upon the ARD method. On the other hand, VIC-KD for both TC-ResNet and XVector can substantially improve upon both robust distillation recipes and the baselines for the clean and under-attack scenarios, thus showing the benefits of the proposed methodology. In fact, VIC-KD with the *Wav2Vec 2.0* / XVector (T1/S2) pair outperforms the baseline by a relative percentual difference of 7.2% on robust accuracy. Similarly, the proposed model outperforms ARD and RSLAD on the attacked scenario (with AutoAttack) by 8.0% and 6.6%, respectively. On the downside, the VIC-KD recipe training time is four times slower than RSLAD. Note that, at the inference stage, the time is the same for all distillation recipes since it depends solely on the architecture of the Student model.

Furthermore, some conclusions can be drawn from the distillation of robust Teachers. First, from the KD experiment, it is crucial to notice that the Student model does not necessarily inherit robustness from the Teacher; hence, specific techniques are needed if the goal is to design adversarially robust models. However, a tradeoff between clean and robust accuracy needs to be decided. For the robust distillation methods, we observe that using robust Teachers, in general, does not improve the final performance of the Student model; thus, for our specific use case on the GSC dataset, unless one already has off-the-shelf robust Teacher models, it is not worth the extra computation of transforming a standard Teacher into a robust one first and then perform the robust distillation.

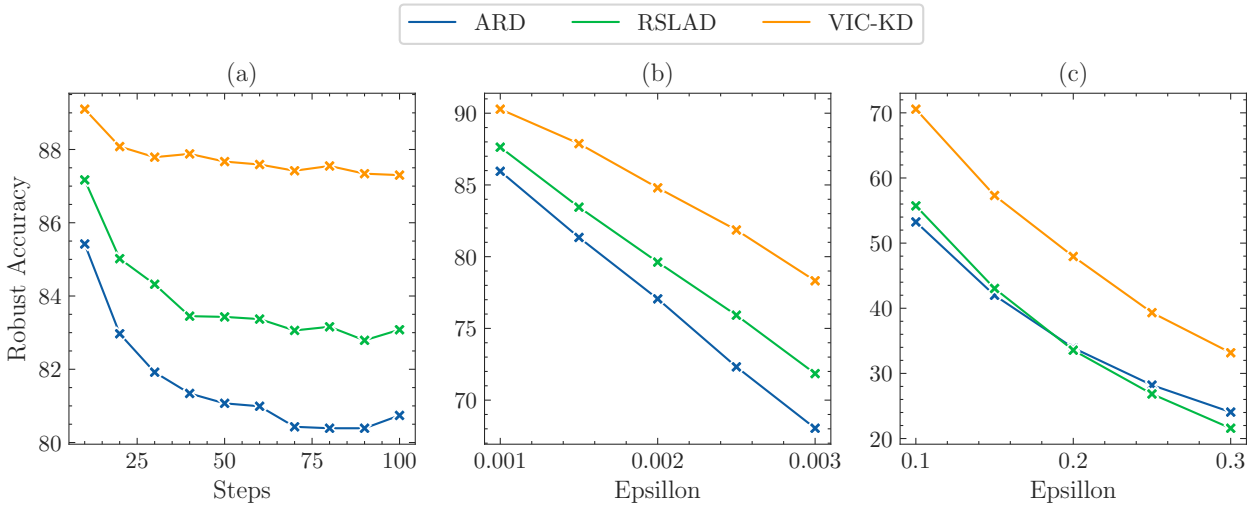


Figure 4.2: (a) Effects of multiple steps on the PGD-ℓ_∞ method. Next we explore the impact of the hyperparameter ϵ that controls the attack strength in PGD$_{40}$ with (b)ℓ_∞-bounded attacks; and, (c) with ℓ_2-bounded attacks. The step is always calculated as $\alpha = \epsilon/4$.

Finally, Figure 4.2 illustrates the sensitivity of the evaluated methods to variations in PGD attack hyperparameters. For instance, we observe that increasing the number of optimization steps (Fig. 4.2 (a)) intensifies the attack strength, resulting in a decline in robust accuracy across all models. Analogous trends are evident when scaling the perturbation budget ϵ for both ℓ_∞ and ℓ_2 constraints. Notably, VIC-KD maintains a distinct performance margin over competing methods even at higher perturbation magnitudes, underscoring its superior capability to learn representations that are invariant to adversarial noise.

4.5.2 Result #2: Effects of Multi-View inputs

Next, we investigate the effect of multi-view (MV) inputs on robust distillation, as shown in Fig. 4.3. Here, we investigate the WavLM/TC-ResNet pair as our Teacher and Student models, respectively. For comparison, a red dashed line shows the baseline robust accuracy of TC-ResNet trained with TRADES without any guidance from Teacher models. As observed in the figure, all distillation methods benefit from multi-view. For instance, ARD and RSLAD, which had a robust accuracy below the baseline, can surpass this landmark after MV. Our proposed VIC-KD method can outperform the baseline with or without MV. However, MV helps us to achieve the best overall robust accuracy.

We hypothesize that the reason behind the performance gain by using MV is two-fold. First, MV induces the Student model to learn perturbation-invariant features from speech signals and to better

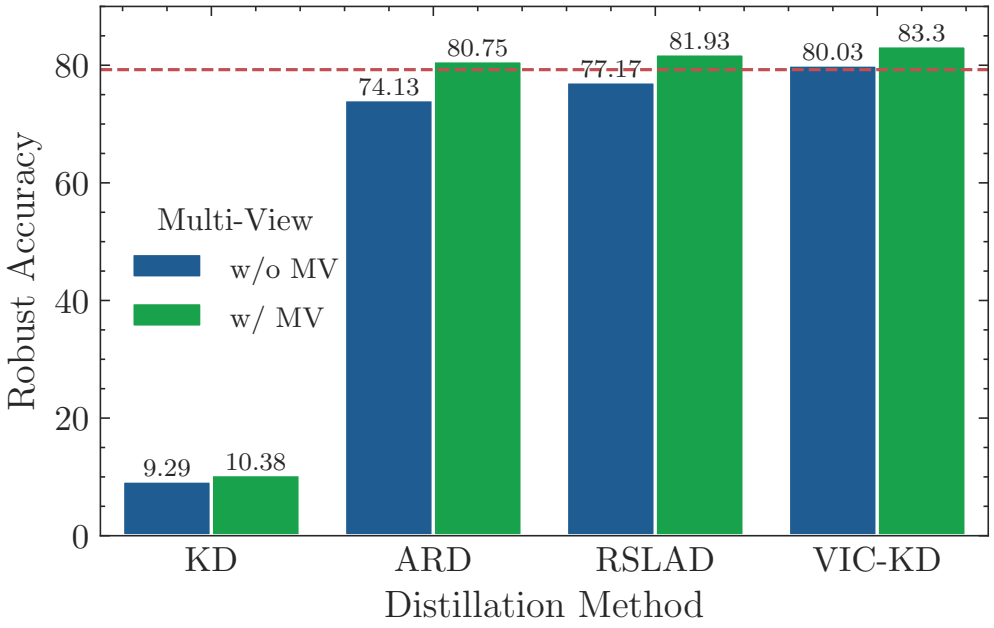


Figure 4.3: Effects of the Multi-View on several distillation Methods for the WavLM / TC-ResNet pair. The dashed line represents the robust accuracy of TC-ResNet via TRADES. The Robust Accuracy is measured when the model is attacked by the AutoAttack method.

disentangle noisy factors from speech features, thus improving generalization. Lastly, other works already have discussed that self-supervised learning methods that employ MV are forcing the model to maximize the mutual information of representations between the two views [58]. We suppose a similar conclusion can be drawn for the KD case, but more studies are still needed.

4.5.3 Result #3: Extending to more classes

Lastly, we further stress the distillation recipes to a more significant number of output classes by using all 35 classes from the GSC dataset. In Fig. 4.4, we show the accuracy results for both clean and AutoAttack-based methods using the WavLM / XVector models as Teacher and Student, respectively. The blue and green dashed lines denote the clean and robust accuracy of the supervised baseline, XVector, trained via TRADES. Similarly to previous findings, KD can surpass the clean accuracy of the baseline, but the robust accuracy is not satisfactory. Among the robust distillation methods, VIC-KD exhibits the overall best performance by achieving the best accuracy and showing the smallest gap between clean and robust accuracies.

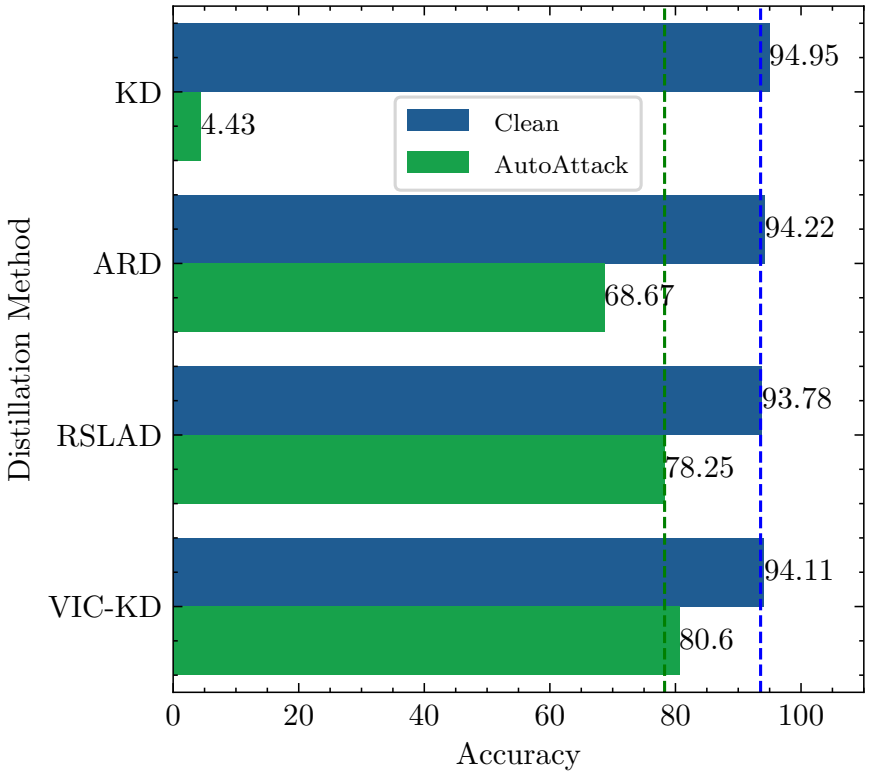


Figure 4.4: Scaling robust distillation to a 35-class GSC dataset and comparing it to clean and robust accuracy baselines. AutoAttack is used to generate adversarial perturbations.

4.6 Conclusions

Here, we propose VIC-KD, a methodology to improve the adversarial robustness of distilled models. Our recipe is two-fold: (1) use multi-view inputs to induce the Student to learn perturbation-invariant features and (2) apply the variance-invariance-covariance regularization (VICReg) to the latent representation of the Teacher/Student model. Experiments on the Google Speech Command dataset, with 12 and 35 classes, show the proposed methodology outperforming state-of-the-art robust distillation recipes on ℓ_∞-bounded attacks (e.g., PDG and AutoAttack). Overall, VIC-KD can better balance the tradeoff between clean and robust accuracy, making this technique a strong candidate for developing and deploying trustworthy speech applications on the edge.

Part II

Applications to Bioacoustics with General-Purpose Audio Encoders

Chapter 5

Leveraging Foundation Models for Acoustic Beehive Monitoring

5.1 Preamble

This chapter is compiled from material extracted from the following publications: Our conference paper at the 2023 IEEE International Workshop on Metrology for Agriculture and Forestry [120], and our journal paper at the IEEE Internet of Things Journal (IEEE IoT-J) [121] (2025).

5.2 Introduction

Honeybees (*Apis mellifera*) are crucial for maintaining ecological balance and biodiversity as proficient pollinators, assisting in the reproduction of various plant species. They also have significant economic value, producing honey and beeswax, and impacting about one-third of global food production [63]. Over the last few decades, however, the decline in the honeybee population size has concerned the scientific community due to its impact on agricultural and general well-being.

In modern beekeeping, Langstroth hives are the standard structure for hosting bee colonies. These rectangular hives consist of stacked boxes, each containing several frames for honeycomb building, as depicted in Figure 5.1. Traditionally, colonies are inspected by the beekeepers, who

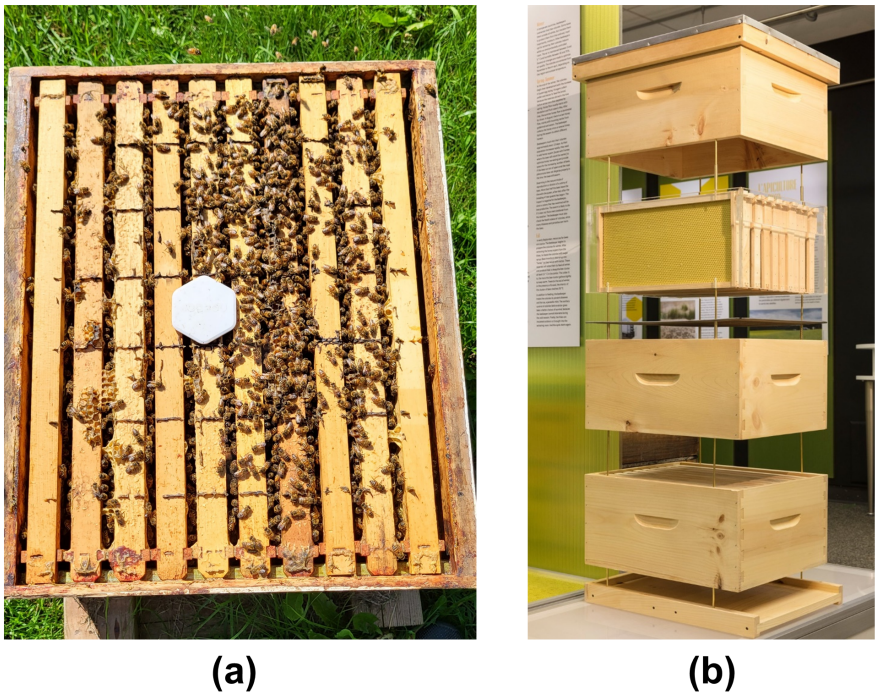


Figure 5.1: Illustrations of a Langstroth hives. In (a), it shows the usual position for the sensor, which is placed over the frames of the brood champer. And (b) the decomposed overall structure. Image reproduced with permission from the authors from [1].

manually examine each hive; this can be time-consuming and laborious for the beekeepers, as well as stressful and disruptive for the bees. Therefore, there is growing interest in automated and remote monitoring systems.

In particular, recent advances in Internet of Things (IoT) research have enabled the deployment of smart beekeeping solutions that integrate non-invasive sensors to monitor hive conditions remotely. IoT-enabled beehive monitoring systems utilize a network of sensors to collect real-time data on key environmental factors such as temperature, humidity, hive weight, and bee activity via image and/or audio. In particular, the acoustics within the hive can correlate with vital measurements, such as swarming [122], queen bee presence [123], health measurement [124], and identifying bee species [125].

In practice, acoustic monitoring systems for beehives need to have low computational and storage footprints, allowing for long battery life. Many apiaries are also located in remote regions, where internet connectivity may be limited, meaning that most of the processing needs to be performed locally (i.e., at the “edge”). To this end, self-supervised learning has emerged as a promising solution.

In addition, data scarcity is a serious challenge in bioacoustics, as labeled wildlife recordings are difficult to obtain. SSL, however, facilitates few-shot learning, thus making it a crucial technology for bioacoustics research [126].

SSL models have already started to be employed in the bioacoustics field, including beehive monitoring. Recent studies have achieved state-of-the-art results for e.g., bird and whale classification [127, 128]. Many studies, however, rely on fully supervised approaches, where different models are trained for different tasks. Several works have also relied on proprietary datasets (e.g., [65]), and those that use public datasets, often partition the data in different ways, making it difficult to compare the results across models and architectures.

In this chapter, we propose to overcome these limitations and propose a standardized evaluation of the performance of audio representations for acoustic beehive monitoring tasks. Our contributions are two-fold. First, we will discuss a new task, based on a popular dataset, of finding beehive recordings contaminated with human speech, which can hinder the performance of subsequent beehive monitoring tasks [66]. For this, we will leverage both speech and audio encoders, as well as discuss anomaly detection approaches, since our labeled dataset is so small.

Second, given the good performance of audio encoders, we propose an evaluation pipeline (benchmark) that consists of three major components: (1) the UpstreamExpert, a wrapper around an audio SSL model that extracts features; (2) the Featurizer, which aggregates representations from multiple layers of the UpstreamExpert; and (3) the DownstreamBrain, which applies these representations to specific tasks.

For this second part, we summarize our contributions as follows:

1. We propose a unified evaluation framework for acoustic beehive monitoring tasks by standardizing task interfaces, training/testing splits, and hyperparameter configurations. This ensures consistent experimental conditions across methods, enabling fair and reproducible comparisons that isolate model performance from other effects.

2. We evaluate existing SSL models under challenging conditions to assess their robustness and generalization, including out-of-domain test sets, limited data, and class imbalance settings. This setup provides insights into how these models might perform in real-world deployment scenarios.

3. Lastly, we compare models of different sizes, aiming to identify suitable audio encoders for beehive monitoring, balancing performance and hardware constraints.

In summary, our goal is to help beekeeping researchers and practitioners quickly select an adequate starting point for machine learning-based hive analysis.

5.3 Selected General-purpose Audio Encoders

In Chapter 2, we already started the discussion on the differences between general-puporse audio encoders and speech encoders. General-purpose audio encoders are typically pretrained on AudioSet [129] (AS-2M), which includes speech, animal, and urban sounds, among them some bee and wasp buzzing samples. Some models also incorporate LibriSpeech [88] (LS), a 960-hour audiobook dataset. Existing bioacoustic-specific encoders leverage other diverse datasets, including the FSD50K [130], VGGSound [131], and the Xeno-canto[1] (XC) for broader animal sounds and vocalizations. Here, we present a few more models that will be important for our work and that will be evaluated on the new beehive-related tasks described herein.

BYOL-A [132, 76]: This general-purpose audio representation learning system employs self-distillation as its self-supervised learning method. Two augmented views of the same audio excerpt are input to the online and target networks, to minimize the mean squared error between their L2-normalized latent representations. Key components include: (i) mel-spectrogram input; (ii) a fully convolutional encoder without Transformer layers, making it a lightweight model.

SS-AST [77]: Is a model inspired by the Vision Transformer (ViT) [29]. First, the input waveform is converted into a log-mel spectrogram, which is then split into patches. The model is pre-trained using a joint task of Masked Spectrogram Patch Modeling (MSPM) and contrastive learning objectives. Here, the tiny, small, and base model versions are used.

MSM-MAE [133]: This model implements a similar idea to SS-AST. MSM-MAE also splits the input along both time and frequency dimensions to create paths, allowing the model to learn

[1] https://xeno-canto.org/

temporal and frequency relations. In their work, the authors rely only on the masked autoencoder reconstruction objective to pre-train a large ViT model.

M2D [134, 135]: M2D builds on the notion that reconstructing masked patches using all available patches may be suboptimal. Inspired by BYOL-A, it employs a self-distillation objective. Specifically, some patches are processed by the online network, while the target network processes the remaining patches. The training objective is to maximize the agreement between the learned representations from both networks.

BEATs [30]: This model adopts a different pretext task approach. Instead of reconstruction, it uses classification loss for masked input by predicting semantic-rich discrete labels. This involves a multi-step training procedure between the audio SSL model and an acoustic tokenizer. The acoustic tokenizer is initialized as the previous step's audio SSL model. Empirical results show that BEATs achieve state-of-the-art performance across several tasks.

CAV-MAE [136]: The Contrastive Audio-Visual Masked Auto-Encoder (CAV-MAE) is a multi-modal model combining contrastive learning and masked modeling for audio-visual representation learning. Despite being trained with both modalities, its audio encoder alone achieves strong performance. Here, we use only the audio encoder, which follows a ViT-based architecture, to extract representations from beehive recordings.

AVES and BirdAVES [31]: Both are SSL feature extractors based on HuBERT [13]. The architecture comprises a CNN encoder followed by a Transformer encoder, trained to predict semantic tokens from masked audio. AVES is pretrained on AS-2M, FSD50K, and VGGSound, here denoted as *BIO*. BirdAVES extends AVES with a larger Transformer (HuBERT-large) and additional pretraining on the XC dataset for better bird sound representation.

5.4 Self-Supervised Features for Background Speech Detection in Beehive Recordings

In a recent study on the effects of environmental factors for beehive acoustic monitoring systems [66], the authors quantified the effect of rain noise, background beekeeper speech, and traffic noise on an automated hive strength monitoring system. The results indicated that these factors significantly degraded system accuracy, especially when background speech was present. Hence, it is crucial to include a preliminary stage to eliminate such unwanted disturbances. While eliminating the interference caused by, e.g., rainfall and winds could be accomplished by utilizing available weather station data, eliminating speech-related distortions can be more challenging.

Here, we aim to explore the use of SSL to detect human background speech in audio recordings collected within a beehive, a task also known as voice activity detection (VAD). Performance is compared against different baseline methods, such as short-time energy thresholding and the widely-used WebRTC method [137]. Furthermore, we explore the capabilities of such universal representations for unsupervised anomaly detection, eliminating the need for manually labeled data.

5.4.1 Methods and Materials

Dataset

In this study, we used audio data from the Nu-hive open source project [138]. The dataset consists of audio recordings from two beehives in Ancona, Italy. The recordings were made at a sampling rate of 32 kHz. The monitoring was conducted for two days: one day when the queen bee was present and another day when the queen bee was absent (i.e., the queen was dead). A subset was selected and manually labeled for a speech-detection experiment to identify the start and end points of noise events. Figure 5.2, for example, shows a 3-minute excerpt from the Nu-hive dataset with the detected noise events shown via the green curve.

More specifically, 20 minutes of data are used for training and another 20 minutes for testing. As demonstrated in [27], one of the most significant advantages of using SSL models is the feasibility of using a few data points for fine-tuning specific tasks. Each sample in our dataset corresponds to a fixed non-overlapping 250 ms audio segment, and a resampling operation is performed to convert

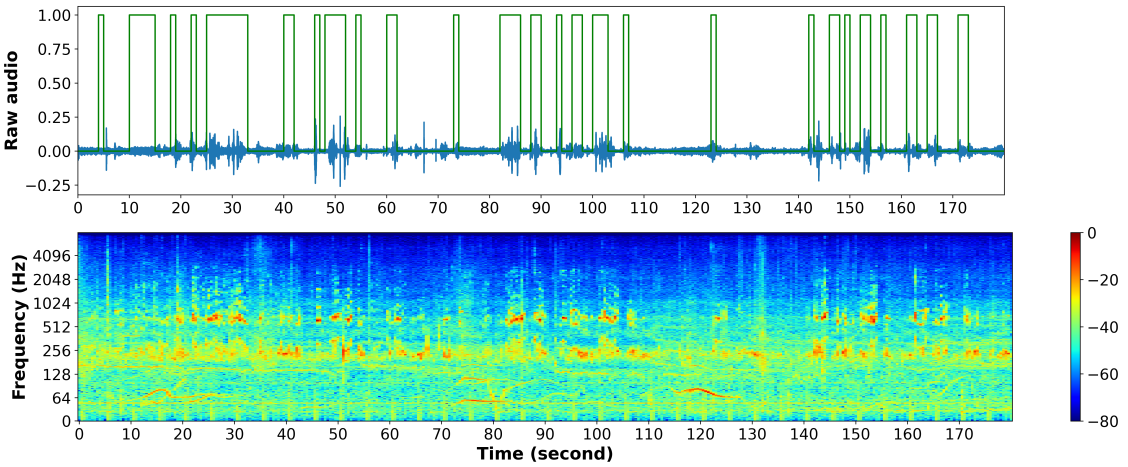


Figure 5.2: A 3-minute excerpt from the Nu-hive dataset, from top to bottom, raw audio signal in blue and ground truth labels in green, and spectrogram.

to a 16 kHz sampling rate. Although this sampling rate is required because the SSL models are trained at this rate, we find it adequate, as speech and bee buzzing have relatively low bandwidth. In the training set, we further split the dataset into training and validation with an 80/20% ratio, respectively. The train, validation, and test datasets have 23.58%, 23.60%, and 4.98% ratios of speech events, and 3545, 1182, and 4736 samples each, respectively.

Baseline methods

Identifying speech in audio segments is a well-established problem in the speech communications literature [139]. We selected four baseline approaches to compare our proposed methodology based on self-supervised features. First, the most trivial case considers a random baseline that outputs the majority class, i.e., all audio segments are bee-buzzing sounds. It helps us understand how other approaches improve speech detection.

Second, we consider short-time energy thresholding, a traditional signal processing technique for voice activity detection. In this approach, we compute the energy ratio of specific frequency subbands where speech utterances are mainly encountered against the total energy across all frequencies in the sample. In our experiments, we set the speech frequency subband and the threshold ratio to $300 - 3000$ Hz and 0.6, respectively.

Next, we explore machine-learning-based methodologies for this problem. WebRTC [137], an open-source initiative, focuses on developing standardized protocols for real-time communication in web browsers and mobile applications. VAD is a crucial element within the WebRTC framework. Foremost, this methodology extracts features such as log energy, zero-crossing rate, and spectral characteristics from small-window frames. Then, a Gaussian mixture model (GMM) classifier is used to determine whether a frame contains speech or not. In this work, we use the Python implementation of the WebRTC VAD [140]. One of the limitations of such implementation is that it works only with 10, 20, or 30 ms window audio. Since the samples in our dataset contain 250 ms, we window the signal and compute the prediction for each segment. Thus, the final prediction is the majority vote across all segments. Here, we choose the 10 ms window because it yields better baseline results.

Lastly, a pre-trained model called VAD-CRDNN, which is part of the SpeechBrain toolkit [60], is explored. A convolutional recurrent deep neural network (CRDNN) is the main component of the model architecture, which jointly exploits the local smoothness and long-range temporal attributes from audio segments. The model is trained on the Libriparty dataset, which contains various simulated acoustic scenes with speech and intermittent noisy sequences, either occurring alone or simultaneously.

Proposed SSL-based Voice Activity Detection

In this work, we develop downstream models based on four SSL models: Wav2Vec 2.0, HuBERT, WavLM, and BYOL-A. Herein, we consider two different architectures depending on the selected feature encoder, as described in the diagram shown in Fig. 5.3. First, when using the Wav2Vec 2.0, HuBERT, or WavLM, the model outputs a 768-dimensional feature vector for every 25 ms of audio. We use the representations from the output of the convolutional network and 12 Transformer layers and combine them using a learnable weighted sum. Next, we use an attentive statistical pooling (SAP) introduced in [141] to aggregate the features on the time domain. For the BYOL-A encoder, on the other hand, since the model already outputs an utterance-level representation on the last layer, we directly feed those 2048 features to the classification head. The classification head is a two-layer perceptron with a ReLU activation, dropout, and batch normalization for all encoders.

Two conditions are explored in our work: (**F**), where we keep the weights from the feature encoder module frozen, only adjusting the weights of the subsequent components; and (**U**), which

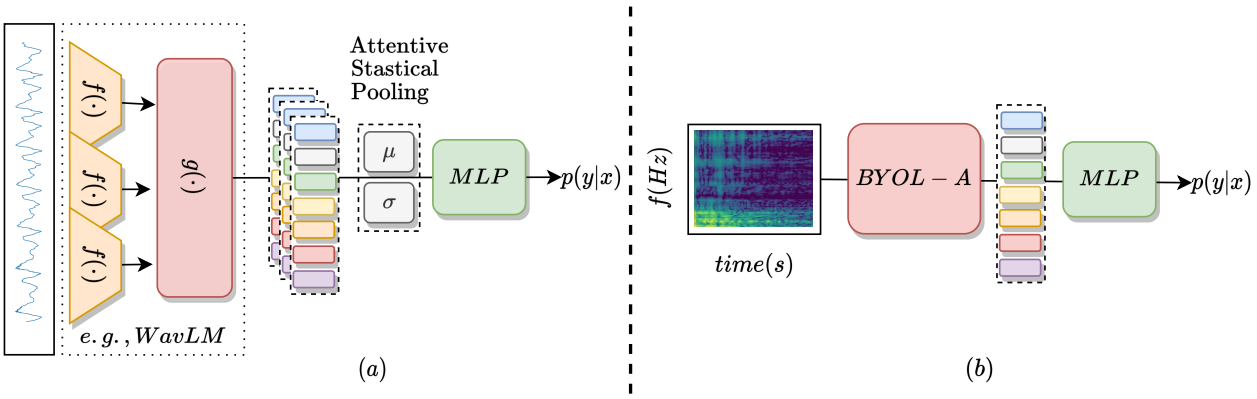


Figure 5.3: Diagram of the proposed system. (a) is the proposed system built on top of speech foundational models; (b) shows how the same objective is implemented with the BYOL-A upstream, which directly extracts an utterance-level representation on the last layer.

allows the gradient to flow through the entire network, updating all the weights. We use the AdamW optimizer for training the networks with a learning rate of 1×10^{-4} and 1×10^{-5} for the (**F**) and (**U**) conditions, respectively. A cosine annealing scheduler decays the learning to a factor during the training. The models are trained on a single RTX 3070 Ti, with 8Gb, for ten epochs. Roughly, it takes 5 minutes to train each network. Lastly, we use the ROC-AUC, Precision, Recall, and F1-score metrics to compare all the models for this imbalanced binary classification problem.

5.4.2 Experimental Results

Classification Performance

Table 5.1 shows the experimental results of our supervised approach. The first four lines are obtained from the baseline models, showing the advantage of the VAD-CRNN approach over the other baselines. However, considering computational cost, we can observe that the short-term energy threshold method is the most suitable.

Next, we can observe performance improvement from our proposed models across all scenarios. Additionally, even though we have few data points for training, using in-domain data for this fine-tuning stage is beneficial to the models. Secondly, unfreezing the weights of the feature encoder can leverage better generalization capabilities when the model has not previously seen bee buzzing sounds. However, for the BYOL-A model, we observe that the frozen and unfrozen conditions yield similar results across metrics, except that we can improve recall and, consequently, the F1 score.

Table 5.1: Experiments for detecting speech sounds on acoustic beehive monitoring, emphasizing our proposed SSL approach. (F) indicates a frozen encoder and (U) unfrozen.

	Method	ROC-AUC	Precision	Recall	F1
Baseline	Random Baseline	50.00	47.51	50.00	48.72
	Short-time Energy	—	62.15	57.36	59.01
	WebRTC	—	49.27	46.65	42.83
	VAD-CRDNN	69.55	60.05	59.73	59.89
Proposed	Wav2Vec 2.0 (F)	73.81	67.53	50.39	49.55
	Wav2Vec 2.0 (U)	79.68	74.95	63.75	67.49
	HuBERT (F)	63.64	66.45	56.76	59.18
	HuBERT (U)	80.89	72.87	66.43	69.03
	WavLM (F)	70.61	57.83	65.78	59.79
	WavLM (U)	**82.88**	66.69	**74.01**	**69.53**
	BYOL-A (F)	81.01	**75.68**	58.12	61.74
	BYOL-A (U)	80.77	74.81	65.76	69.12

From the models pretrained with speech data, Wav2Vec 2.0, HuBERT, and WavLM, we observe that we can consistently outperform the baselines on the ROC-AUC metric. The Wav2Vec 2.0 has the best ROC-AUC and precision for those encoders in the frozen condition. Empirically, we attribute this result to the fact that contrastively trained models can leverage better-disentangled representations, thereby enhancing the separability between speech and non-speech samples.

HuBERT and WavLM, on the other hand, can outperform the baselines with the frozen encoder scenario. However, when allowing the fine-tuning of the encoder, we can achieve the best overall performance with the WavLM feature encoder. We hypothesize that, since WavLM is a robust model, it learns noise-invariant features for speech signals and can effectively detect speech against other sounds.

As discussed before, BYOL-A has the best performance in the frozen scenario, followed closely by the unfrozen conditions. We attribute this behavior to the fact that BYOL-A (5M parameters, against the other models with $\approx$ 95M) has limited capacity compared to the other encoders; thus, extracting the maximum performance of such a model with simple linear probing is difficult. In future work, we would like to investigate if extracting the representations from intermediate layers of the model and frame-level features instead of utterance-level ones can help maximize the information input to the downstream model.

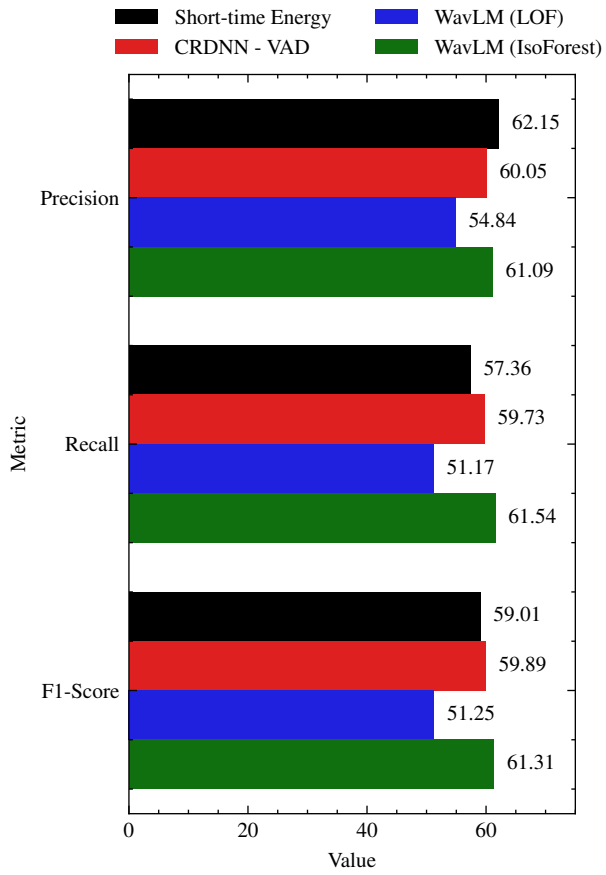


Figure 5.4: Comparing unsupervised approaches built on top of SSL-based representations against strong baselines.

Unsupervised Anomaly Detection

To further explore the capabilities of self-supervised audio representations, we evaluate their performance in an unsupervised anomaly detection scenario. Since WavLM had the best overall performance in the supervised setting, we selected this model as our feature generator. Herein, we directly sum all the intermediate representations of the model and average across time to get a single global representation, a vector of size $B \times 768$, where B is the test set size. Next, we train two popular algorithms for this task of unsupervised anomaly detection: Local Outlier Factor (LOF) [142] and Isolation Forest [143].

Figure 5.4 shows the obtained results with our approach. We observe that, even in an unsupervised scenario, using SSL-based representations can leverage good performance in identifying if background speech has occurred in an audio segment. In particular, the Recall and the F1-score perform best

Table 5.2: Comparison of related work and our approach. Column "Type" refers to Benchmark or Evaluation studies. "# Models" and "# Tasks" indicate the number of SSL models and tasks evaluated. "Bio?" marks bioacoustic focus, and "ABM" indicates inclusion of at least one Acoustic Beehive Monitoring task.

Work	Type	# Models	# Tasks	Bio?	ABM?
SUPERB	Benchmark	15	10	✗	✗
HEAR	Benchmark	16	19	✗	✓
Idiap	Evaluation	7	3	✓	✗
Ours	Evaluation	14	4	✓	✓

compared to robust baselines, showing that the proposed methodology is more sensitive to identifying voice activity.

5.5 Benchmarking Audio Representations for General Acoustic Beehive Monitoring

The practice of benchmarking and/or evaluating models across a diverse set of audio/speech tasks has become commonplace. Representative examples include (i) the SUPERB [32] benchmark for speech; (ii) the Holistic Evaluation of Audio Representations (HEAR) [144] Challenge, covering speech, music, and environmental audio applications, and (iii) the Idiap Human-Animal Assessment [145], where speech and a bioacoustic encoders were tested across three tasks: identification of mammal marine species, call-type classification of infant marmosets (*Callithrix jacchus*), and dog vocalization classification.

As can be seen, benchmarking of popular general-purpose and bioacoustic audio encoders for beehive monitoring tasks, however, is not as advanced. Our evaluation system aims to fill this gap and shares some evaluation features with these three abovementioned audio benchmarks. In particular, compared to the Idiap benchmark, our work goes deeper into acoustic scenes specific to beekeeping contexts, and explores general-purpose audio encoders, since speech encoders might not be best suited for the frequency range of bee sounds. Table 5.2 summarizes the key main differences between existing and the proposed benchmark, including if they include a bioacoustics task and an acoustic beehive monitoring (ABM) task.

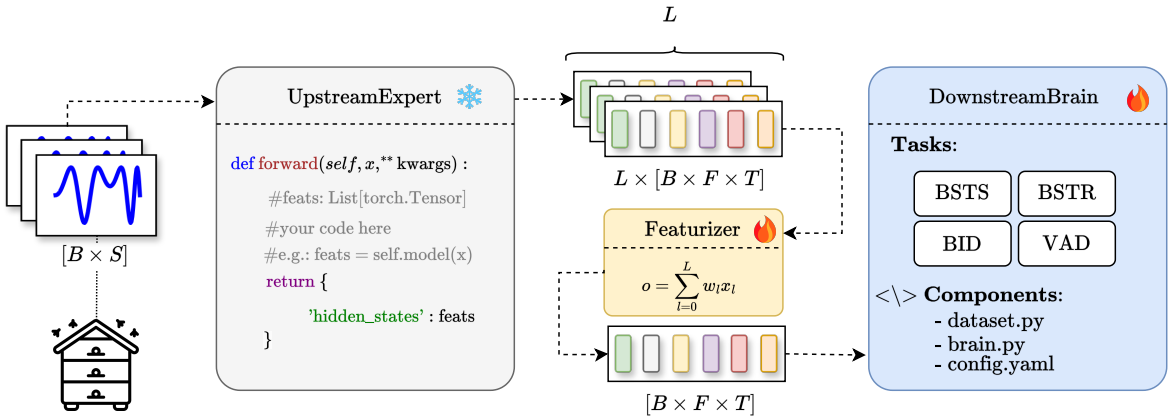


Figure 5.5: Schematics of the proposed evaluation system highlighting the three main components: The UpstreamExpert, the Featurizer, and the DownstreamExpert.

Figure 5.5 presents a high-level schematic of the developed benchmark framework. The first step is audio acquisition, where microphones capture data from the beehive for processing on a remote server or locally. The audio is then input to the UpstreamExpert, which extracts a list of representations. The Featurizer, a learnable weighted sum, transforms these representations into a single tensor. Finally, the DownstreamBrain handles training and testing to solve tasks (e.g., detecting the queen bee's presence) and outputs performance metrics. The code is written in Python using the PyTorch framework [146] and the SpeechBrain toolkit [60] to handle audio tasks. In the next subsections, we dive into each of these components. The code can be found at the following GitHub link: `https://github.com/MuSAELab/hunee-b`.

5.5.1 Upstream Components: Self-supervised Audio Encoders

The first component presented is the audio encoder. Our system is designed for easy extension with various foundation models. To assess a new model's performance, we encapsulate it within an UpstreamExpert container, which should include routines for loading pretrained weights and forward samples to the model. The output from the forward function must be a dictionary with a single key, 'hidden_states', containing a list of extracted representations. If the list has more elements than the output from the last neural network layer, each tensor must have the same shape, $[B \times F \times T]$, where B is the batch size, F is the embedding dimension, and T is the time dimension. Additionally, the design of smaller upstream models will be encouraged and will receive more points in the overall score, as detailed later.

5.5.2 Featurizer Component: Attending different representations

In the context of deep neural networks, each layer extracts its own set of features. Typically, early layers capture simpler features, while deeper layers extract high-level and abstract factors of variation [25]. Consequently, features from different layers can be crucial to address various downstream tasks.

Similar to SUPERB, we introduce the Featurizer, a component that applies a learnable weighted sum to combine representations from multiple layers, allowing for a more flexible representation. From an upstream model with L layers, we extract a set of latent representations $X = [\boldsymbol{x}_1, ..., \boldsymbol{x}_L]$ and create a weight vector $W = [w_1, ..., w_L]$. Initially, W is a vector of ones. During downstream training, W is learned and normalized via a softmax function. The output is a single tensor $\boldsymbol{o} = \sum_{l=0}^{L} w_l \boldsymbol{x}_l$. This mechanism allows the model to learn which layers contain the most relevant features for a specific downstream task, rather than relying solely on the final layer.

5.5.3 Downstream Components: Datasets, models, and evaluation

The last fundamental pieces of the proposed benchmark are the downstream components. The downstream is a set of modules responsible for taking the extracted representations and applying them to a specific task. Since the upstream models are trained in a self-supervised manner, probing the representations on multiple tasks is a way to ensure and evaluate whether the models are learning to extract "universal" and meaningful features from the recordings themselves. Here, four downstream tasks have been selected for evaluation: Beehive States (BSTS), Beehive Strength (BSTR), Buzzing Identification (BID), and Voice Activity Detection (VAD). For reproducibility and future works, each of these tasks are implemented and configured via three source files:

1) *config.yaml*: This configuration file contains essential information for the tasks, including dataset paths, hyperparameters, neural network architecture, optimizers, and more.

2) *dataset.py*: This Python script encompasses both data preparation steps and SpeechBrain-specific data I/O pipelines. Initially, JSON files are generated to store meta-information about the dataset being fed into the model. Subsequently, a loader pipeline is constructed using the *DynamicItemDataset* class from SpeechBrain. This class helps with the abstraction of collate

functions for audio of varying lengths and enables the creation of audio batches for processing by the models.

3) *brain.py*: This Python script implements the DownstreamBrain class, derived from the *Brain* class in SpeechBrain. It orchestrates the downstream task, including functions for the forward and backward processes, optimization steps, and the entire training loop.

Although not designed to be a benchmark tool with a leaderboard, our code base can support the case where a user intends to evaluate a new model. In this setting, the Python script should not need any modifications, only some fields of the *config.yaml* file to choose the batch size and number of epochs. Furthermore, if a researcher wants to extend the evaluation to a new task, those files serve as an opinionated guide for development.

In the following paragraphs, we provide a detailed explanation of each downstream task, its objectives, and the intuition behind it. Our system aims to assess the generalization capabilities of the upstream models in generating meaningful features for acoustic beehive monitoring. Hence, we avoid using large models for discriminative tasks, typically employing a simple multilayer perceptron (MLP) with two layers at most. Additional details to reproduce our experiment results, such as downstream hyperparameters, are briefly described below. Note that all of the configurations are inside the GitHub repository, where users can follow the instructions to run the codes.

Beehive States (BSTS) Given a beehive recording, this task aims to identify whether the queen bee is present in that sample. Identifying the absence of the queen bee is fundamental to beehive management [64]. The queen bee is essential for the colony's stability, from reproducing duties to increase the number of worker bees, to pheromone production to maintaining social harmony. If a queen bee is dead, two possible outcomes can happen: (i) the generation of a new queen through natural rearing [147] or (ii) a swarming phenomenon takes place, where the hive splits into two or more groups that will leave the current hive to start new colonies elsewhere [148]. With swarming, the chances are uncertain that the remaining and the new colonies will survive.

Audio monitoring of beehive states is a non-invasive method that can help beekeepers rapidly introduce a new queen into the hive, avoiding the catastrophic death of entire colonies. Some works in the literature have already relied on machine learning algorithms to classify states, showing

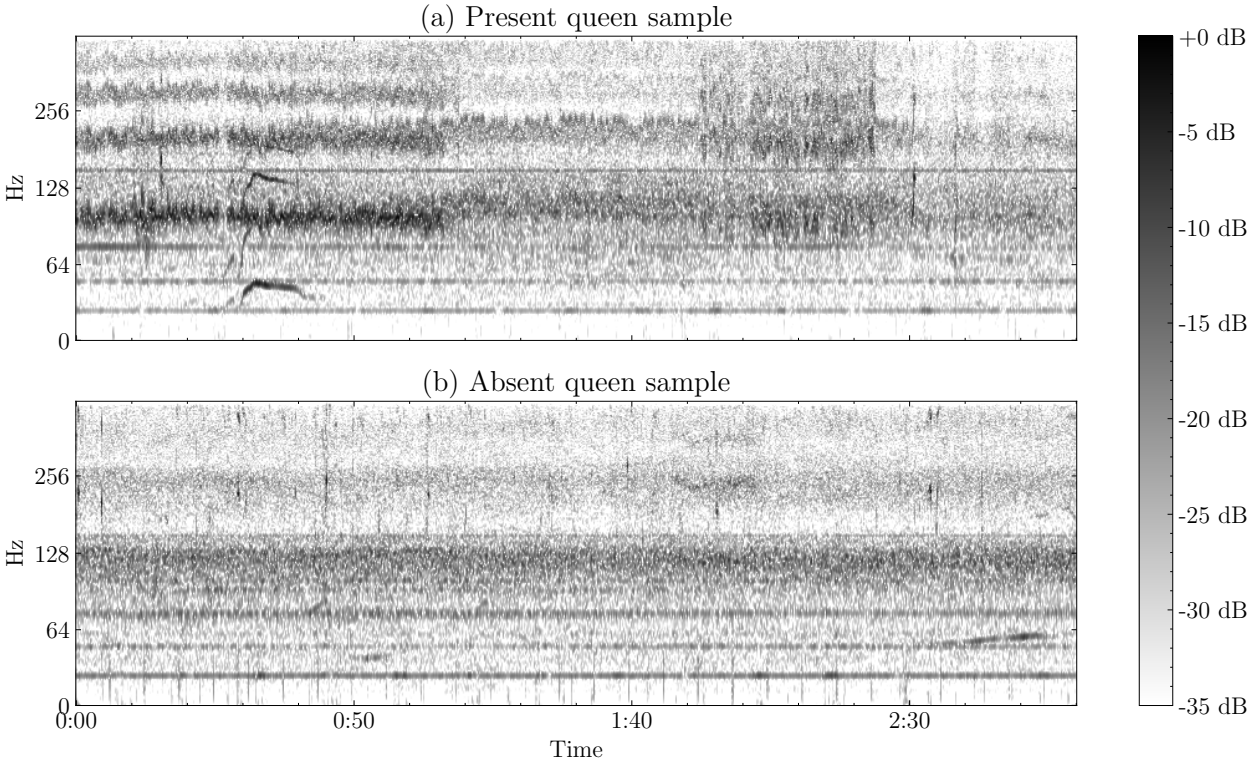


Figure 5.6: Spectrograms of 3-minute excerpts from the NU-Hive dataset. The audios are taken from the same Hive in similar times during the night. (a) is a sample where the queen was present in the hive, and (b) is a sample when the queen was dead.

the importance of frequency domain features [149, 150]. Figure 5.6 illustrates the differences in spectrograms from audio collected from hives with and without a queen bee.

The audio data used for this task is based on the NU-Hive project [151]. One key challenge of this task is that, similarly to [152], we use a hive-independent setting. This means that the training and testing sets are from different hives, requiring the design of algorithms that can effectively handle distribution shifts. The goal of this task is to measure whether current machine-learning algorithms can address this issue and capture factors related to bee buzzing sounds while remaining invariant to other acoustic properties, such as hive reverberation, environmental noises, and microphone positioning.

The NU-Hive dataset was recorded in two beehives in Ancona, Italy, on two separate days. The first-day recording includes the presence of the queen bee, while the second-day recording occurred after the queen had died. Originally, the recordings were made at a sampling rate of 32 kHz; however, these have been resampled to 16 kHz to ensure compatibility with most existing audio encoders. Lastly, the BSTS task is treated as a binary classification problem, and the ROC-AUC metric is

Table 5.3: Detailed description of the data used for each Downstream Task of our system. Columns "Hive Ind.?" and "TVT #samples" describe whether a hive-independent setup is used and the number of samples in the train, validation, and test sets, respectively.

Task	Type	Labels	#Minutes	Hive Ind.?	TVT #samples	Apiary Location
BSTS	Binary Classif.	{NoQueen, Queen}	5760	Yes	(5120//640//5760)	Italy
BSTR	Regression	$y \in [10, 30]$	2887	Yes	(6910//2317//2320)	Canada
BID	Multiclass Classif.	{Bee, Cricket, Noise}	509	Yes	(9000//2746//3508)	USA
VAD	Binary Classif.	{NoSpeech, Speech}	40	No	(898//300//1198)	Italy

used. Initially, each sample in the dataset had a duration of 10 minutes, but we split this into non-overlapping chunks of 30 seconds. Additional details about the number of samples, duration, and other information can be found in Table 5.3. Attentive Statistical Pooling [153] is used in the tensor input to aggregate over the time-dimension. The downstream model comprises a simple 2-layer MLP. The RMSProp [154] optimizer is used in this task, and we train for 10 epochs with a batch size of 4 samples. Data augmentation is employed through SpecAugment [155], where we mask both frequency bands and time segments during training to prevent overfitting.

Beehive Strength (BSTR) The next task is to predict the strength of a beehive. Similar to other studies [65, 66], we use the number of frames of bees as a proxy for beehive strength. Traditionally, experienced beekeepers inspect each frame visually to assess colony size and overall health. An 'x' number of frames of bees indicates that 'x' frames in the hive have over 80% coverage by honey bees. Larger 'x' indicates greater hive strength. Measuring strength, however, is an invasive process, requiring the opening of beehive boxes and removal of frames. This can be time consuming and laborious for the beekeeper and stressful and disruptive for the bees. The BSTR task aims to predict the number of frames of bees inside a beehive using only the audio signals, allowing scalable and non-invasive monitoring of the colony's health.

For the BSTR task, we utilize the UrBAN [156] dataset. This dataset includes over 16,000 hours of audio recordings from ten beehives monitored in an urban rooftop apiary in Montreal, Canada, during 2021 and 2022. Each beehive follows the standard Langstroth design, with up to three boxes per hive and 10 frames per box, resulting in a possible output range of 0 to 30 frames of bees. The microphone is positioned on the center frame of the brood chamber (the bottom-most box) for recording. The interested reader is referred to [156] for more dataset details.

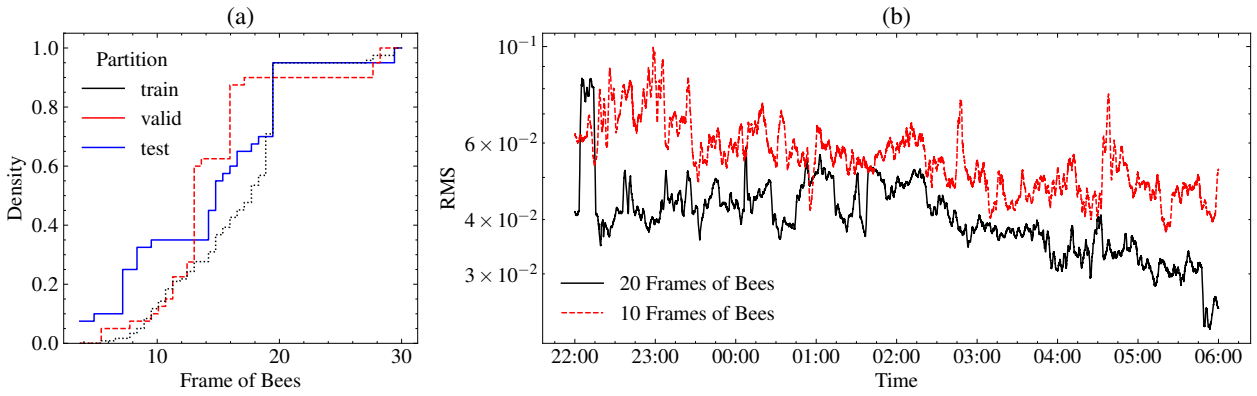


Figure 5.7: Data description for the BSTR task: (a) Cumulative density function of labels per set, (b) RMS distribution over the night for two recordings of the same hive. The black line represents the beginning of summer with 20 frames of bees, and the red line represents early fall with half the colony size.

For this dataset, several adaptations to the original UrBAN dataset have been implemented. First, the ten hives were divided into training, validation, and test sets with a 6:2:2 ratio. Beekeepers annotated the dataset approximately every two weeks, reporting the number of frames of bees for each hive. To infer labels for recordings between inspections, we applied exponential interpolation. Furthermore, we selected recordings taken between 10 p.m. and 6 a.m., when bees are typically inside the hive, maintaining a steady temperature through kinetic energy. Additionally, since each recording is 15 minutes long, we divided them into non-overlapping 15-second segments.

Figure 5.7 provides details on label distribution and the root-mean-square audio power from the same hive during two different periods of the year. The hive started summer with 20 frames of bees and had half that number by mid-September. Note that the RMS is lower when the hive has more frames, as adding frames is done during honey flow periods. This is likely due to the sound absorption properties of the wax and honey present. As such, much like far-field speech applications, algorithms must account for both bee buzzes and hive acoustics to analyze the data accurately. Here, model performance is evaluated using the mean absolute error (MAE) metric. The downstream model consists of a single linear head that directly predicts the number of bees' frames. We leverage the Adam [157] optimizer to minimize the mean square error (MSE) loss function during 15 epochs with a batch size of 8 samples.

Buzzing Identification (BID) In this task, audio recordings are classified into three categories: bee buzzing, cricket chirping, or ambient natural noises. This task uses the BUZZ3 dataset [158],

comprised of recordings from two different apiaries in Utah, USA. This task is referred to as buzz identification (BID), an essential component for "in-the-wild" acoustic beehive monitoring. The most common use of the BID task involves applying supervised machine learning algorithms to beehive recordings, which often contain a variety of natural sounds beyond bee buzzing. Additionally, BID can serve as a pre-processing step to filter recordings and remove artifacts, thus determining their suitability for subsequent analysis in the processing pipeline.

Effectively disentangling bee buzzes from other natural acoustic elements can demonstrate the algorithm's ability to focus only on relevant regions of the audio signals. Bee buzzing sounds can be produced through different mechanisms, such as by gross body or wing movements, muscle contractions, and even pressing the thorax against another bee [64]. On the other hand, cricket chirping is produced by simply rubbing their legs or wings together [159]. Lastly, ambient natural noises include recordings without bee buzzing or chirping, such as rain, vehicles, or microphone noise. These differences suggest the three classes are separable.

This dataset is useful to assess the generalizability of the universal representations. First, unlike most of the other datasets, the microphone is not placed inside the hive, but instead, on top of the outcover in the Langstroth beehive box. Audio recording is done using a custom-built Raspberry Pi setup. Moreover, two different races of bees are recorded: the *Apis mellifera ligustica* (Italian) and the *Apis mellifera carnica* (Carniolan). These configurations enable tests to see how well algorithms generalize to different microphone positions, hardware equipment, and bee races. The task at hand involves a 3-class classification problem. The standard accuracy is the evaluation metric. On the downstream architecture, here, again, attentive statistical pooling is applied to aggregate information along the time dimension. The downstream model consists of a simple 2-layer MLP, similar to the one used in the BSTS task. Data augmentation is performed using SpecAugment, and the RMSProp optimizer is utilized. The models are trained for 20 epochs with a batch size of 32 samples.

Voice Activity Detection (VAD) The last task is to determine whether human speech is present or not in a beehive recording. For this task, the NU-Hive dataset is used, following the same setup as [120] (Section 5.4 of this thesis). The VAD is also treated as a binary classification task, and the F1-score is used as the figure-of-merit to account for the data imbalance. For this task, there are two main challenges: limited labeled data for training the downstream model and highly imbalanced data.

The downstream model includes an attentive statistical pooling layer aggregating information over time, followed by a 2-layer MLP. The optimization process is handled by the AdamW optimizer [160], and we train the model for 25 epochs with a batch size of 32 samples. In order to deal with the dataset imbalance, the binary cross-entropy loss uses a rescaling weight of 8 (i.e., the loss of a positive sample is eight times greater than the negative one).

Overall, all datasets present a great variety in recording conditions. The NU-Hive dataset, for instance, contains recordings with background speech, traffic, and bird sounds. The microphone placement is another factor of variation from dataset to dataset. Our work focuses on evaluating general-purpose self-supervised audio encoders, which are designed to learn representations that are, ideally, robust to variations in acoustic conditions, including microphone placement. These models benefit from pretraining on diverse datasets, which helps improve their invariance to such factors. Additionally, standardizing microphone placement across the community is challenging due to variability in hive designs, monitoring hardware, and environmental constraints across apiaries.

5.5.4 Summarizing the results with a single score

Lastly for our evaluation setup, a general score across all downstream tasks is computed to assess the global performance of the upstream models. As each task has its own figures-of-merit (e.g., accuracy and MAE), simple averaging across tasks is impossible. We leverage a similar formulation to [80], here represented in equation 2.3. However, given the task at hand, smaller upstream models are encouraged to allow for low-resource edge applications typically seen with beehive monitoring systems. This can also reduce costs, thus lowering the barrier of entry for smaller beekeeping teams to try out SSL models in their acoustic beehive monitoring systems. Herein, we denote the number of parameters of an SSL model as $\#P$. Therefore, given a set τ of downstream tasks and the model sizes, $\tau = \{\text{BSTS, BSTR, BID, VAD, \#P}\}$, the final score is computed as:

$$\text{Score}(u) = \left\lceil \frac{1000}{|\tau|} \sum_{t \in \tau} \frac{s_t(u) - s_t(\text{base})}{s_t(\text{SOTA}) - s_t(\text{base})} \right\rceil, \tag{5.1}$$

where u is the upstream model being evaluated (e.g., M2D [134]), $s_t(u)$ is the metric value for task t or number of parameters from model u (e.g., $s_{\text{BID}}(\text{M2D}) = 98.09$ from Table 4.4). Following this notation, $s_t(\text{SOTA})$ and $s_t(\text{base})$ represent the best and worst metrics achieved in the given task t, respectively. More intuitively, we scale each column to the [0, 1] range, sum the values, and rescale

by the number of tasks. The final score ranges from 0 to 1000, with higher values indicating better model performance.

5.5.5 Baselines Models

Here, we briefly describe the fourteen models used in this work. In addition to the eleven audio foundation models already presented in Section 5.3 (SS-AST was used in three configurations: tiny, small, and base, along with BirdAVES in base and large), we also explore three fundamental/conventional upstream models based on time-frequency representations: Filter Banks (FBanks), Spectrogram, and Mel-frequency cepstral coefficients (MFCC). These representations have been commonly used in the bioacoustic beehive monitoring literature. Additionally, users can choose to select the inclusion of delta and delta-delta features (first and second-order derivatives) in the UpstreamExpert.

5.5.6 Experimental Results and Discussion

General accuracy across tasks

Table 5.4 provides a comprehensive evaluation of the 14 tested SSL models across tasks and the overall score taking model size into account. The evaluations were run ten times using a deterministic seed for reproducibility. The numbers listed in parentheses in the Table report the average and standard deviation obtained across these 10 iterations. The scores are computed with the average value of each task, providing a robust overview of the models' performance.

As can be seen, the beehive state classification task poses a significant challenge for neural-network-based models due to the out-of-distribution nature of the task. Test samples, originating from a different beehive than the training one, introduce variations that are difficult to capture by the models and the limited available data. Interestingly, only four out of eleven SSL models outperformed a simple signal representation technique (e.g., Filterbanks), underscoring the complexity of the task. This result suggests that handcrafted features, which have been traditionally used in bioacoustics research, still provide competitive baselines, especially in out-of-domain scenarios where deep models struggle with generalization. Furthermore, these findings emphasize the need for developing self-

Table 5.4: Main evaluation results. Bold represents the best and underline the second best average results.

Upstream Information			Downstream Tasks				
Method	Pretraining Data	#P ↓	BSTS ROC-AUC ↑	BSTR MAE ↓	BID ACC ↑	VAD F1-Score ↑	Score
DSP-based Features							
FBanks	—	0	90.04 (± 2.40)	10.84 (± 0.29)	85.39 (± 9.24)	44.53 (± 8.46)	530
Spectrogram	—	0	86.75 (± 2.88)	10.95 (± 0.32)	94.49 (± 5.02)	51.10 (± 7.30)	607
MFCC	—	0	71.42 (± 8.94)	7.22 (± 3.83)	97.95 (± 1.22)	30.28 (± 5.90)	533
General-Purpose Audio Encoders							
BYOL-A	AS-2M	5M	63.81 (± 7.61)	3.14 (± 0.21)	98.18 (± 0.91)	63.07 (± 8.83)	733
SS-AST (Tiny)	AS-2M + LS	6M	90.24 (± 2.43)	4.44 (± 0.28)	91.66 (± 2.42)	65.13 (± 1.97)	832
SS-AST (Small)	AS-2M + LS	23M	92.70 (± 2.05)	4.45 (± 0.16)	96.61 (± 2.03)	72.43 (± 0.96)	**910**
SS-AST (Base)	AS-2M + LS	89M	**93.91** (± 1.44)	4.11 (± 0.22)	74.22 (± 5.95)	72.19 (± 2.32)	706
MSM-MAE	AS-2M	92M	89.33 (± 1.81)	4.05 (± 0.13)	97.24 (± 0.81)	73.27 (± 1.16)	863
M2D	AS-2M	85M	89.22 (± 3.45)	3.89 (± 0.11)	98.09 (± 0.66)	**74.49** (± 1.37)	883
CAV-MAE	AS-2M	85M	83.37 (± 7.72)	3.42 (± 0.22)	93.60 (± 2.50)	69.26 (± 3.31)	797
BEATs	AS-2M	90M	75.22 (± 4.30)	**3.05** (± 0.12)	99.44 (± 0.18)	72.82 (± 1.06)	811
Bioacoustic-based Audio Encoders							
AVES	BIO	94M	90.48 (± 1.80)	3.35 (± 0.16)	**99.46** (± 0.08)	74.42 (± 0.95)	**910**
BirdAVES (Base)	BIO + XC	94M	85.32 (± 3.64)	3.39 (± 0.17)	98.22 (± 0.53)	68.59 (± 2.93)	838
BirdAVES (Large)	BIO + XC	315M	88.98 (± 1.04)	3.56 (± 0.10)	99.02 (± 0.20)	72.72 (± 1.61)	743

supervised representations that explicitly account for domain shifts of acoustic properties, either through domain adaptation techniques, or data augmentation techniques.

On the other hand, for the beehive strength prediction task, all tested SSL models achieved substantially better results relative to the simpler baseline models. Like the previous test, this task is also hive-independent, but the test samples use the same recording device placed in a similar location inside the Langstroth hive, which have all been located in close proximity in the rooftop apiary. As such, data distribution shifts may have been less pronounced. For this task, the BEATs encoder achieved the lowest MAE. Small-capacity models, such as BYOL-A, also demonstrated strong performance, closely followed by the bioacoustic-based encoders. This suggests that effective models for beehive strength estimation must disentangle semantic content from broader acoustic properties, such as reverberation. Intuitively, the number of frames should influence the hive's overall acoustic profile. Specifically, reverberation within the Langstroth hive is affected by structural elements such as the amount of honey stored in the frames and the number of stacked hive boxes, both of which alter how sound propagates and decays inside the hive. Therefore, an effective encoder must capture these nuanced acoustic variations while remaining invariant to other factors.

Similar conclusions can also be drawn for the buzzing identification task, where most models achieved high accuracy, often outperforming the conventional baselines. Here, bioacoustic-based encoders achieved the best performance, likely because they were trained on diverse animal vocalizations and body sounds and could better capture fine-grained spectral patterns. These findings indicate that specialized pretraining on bioacoustic data may be particularly beneficial for tasks involving fine-grained species or behavior classification, reinforcing the value of domain-specific representation learning in bioacoustics.

Lastly, for the VAD task, it can be seen that the larger models tend to outperform smaller ones and substantially improve over conventional baselines. For the final score, figures-of-merit and model sizes are taken into account. Tied in the first place, the two better-performing models correspond to the SS-AST model (small) and the AVES model. Although the AVES usually performs better across the task, its score is penalized compared to the SS-AST (small) model because the model size is almost 4 times larger. The M2D encoder has the second-best performance in our evaluation. Interestingly, small models that could be considered for lightweight edge-processing, such as SS-AST (Tiny), are not so far in performance, showing better performance than the largest encoder. Altogether, these models were trained on diverse datasets, which may enable better capture of the complex time and frequency patterns seen with bee buzzes, thus allowing for greater generalizability across tasks, as demonstrated in our evaluation.

Continually pre-training with buzzing sounds

Lastly, additional experiments were conducted during the pre-training phase of the BYOL-A model using the UrBAN dataset (specifically the 2021 subset without the empty-hive data) to test the impact of including bee buzzing data during the encoder pre-training. Figure 5.8 shows the results obtained for two different configurations: (a) original BYOL-A trained only on AudioSet, and (b) continued pre-training of the model with the bee buzzing dataset for ten epochs using a small learning rate. It is important to emphasize that to avoid data leakage issues, in configuration (b), all samples from the UrBAN dataset used in the BSTR task were removed.

The obtained results demonstrate that continual pre-training with the bee-buzzing dataset (i.e., configuration b) improved performance in three out of four downstream tasks. These results suggest that including beehive sound data during training can be invaluable. However, training BYOL-A

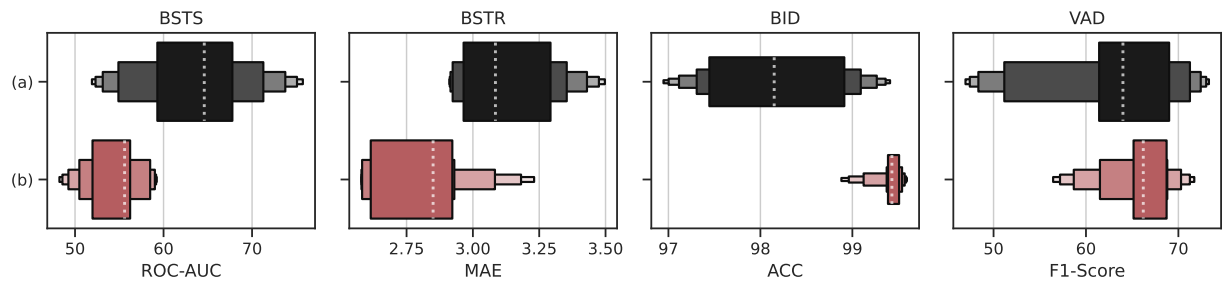


Figure 5.8: Pre-training experiments with (a) AudioSet-only and (b) AudioSet and the UrBAN 2021 from scratch using only bee-buzzing data did not yield promising results. This indicates that there is still significant room for improvement and for the development of dedicated upstream models for beehive monitoring tasks, leveraging large unlabeled bioacoustic datasets.

Limitations

While the present work represents the first attempt to provide a comprehensive evaluation for acoustic beehive monitoring, we acknowledge that a wider range of tasks would be valuable for this work. While the literature has reported uses of automated hive monitoring for swarming detection [161, 162], varroa mite identification [163], bee species identification [164], and specialized bee sounds, such as quacking [165] and tooting [166], the availability of publicly available datasets remain scarce. Typically, datasets are collected and labeled by specific research teams or companies and are often subject to restrictive licenses or become unavailable after several years. The work here has leveraged all the open datasets (and their corresponding tasks) that were found by the authors. Notwithstanding, to help further advance the field of bioacoustics, we make all the codes available to allow researchers with access to proprietary databases to evaluate and benchmark their systems.

5.6 Conclusions

In this Chapter, we introduced a comprehensive benchmarking and evaluation framework for self-supervised audio models in IoT-enabled acoustic beehive monitoring. In particular, we evaluated fourteen SSL models across four key tasks: beehive state classification, strength prediction, buzzing identification, and voice activity detection. We show that SSL models trained on diverse audio datasets, including bioacoustics corpora, provide a strong foundation for non-invasive acoustic

beehive monitoring. While the tested models performed well across most tasks, out-of-distribution generalization remains a challenge, particularly for beehive state classification, where handcrafted features still serve as competitive baselines.

Overall, our findings highlight the importance of model size and domain adaptation, as compact models like SS-AST (Tiny) achieved strong results, making them suitable for low-power edge computing applications. Additionally, continually pretraining with bee buzzing data improved performance, emphasizing the need for domain-specific adaptation in SSL models. By releasing the source code of our system, we aim to drive further research in bioacoustic SSL models and scalable, non-invasive beekeeping technologies.

Chapter 6

BioME: A Bioacoustic Modulation-Aware Neural Encoder

6.1 Preamble

This chapter is compiled from material extracted from [167], currently under review in the IEEE Journal of Selected Areas in Sensors (JSAS).

6.2 Introduction

In the previous chapter, we discussed the critical role of honeybees in ecological stability and food security, benchmarking how popular acoustic foundation models can be leveraged for various beehive monitoring tasks. However, this problem extends beyond honeybees; biodiversity as a whole is facing an unprecedented decline due to climate change and anthropogenic actions [21]. To mitigate this problem, passive acoustic monitoring (PAM) has emerged as a powerful methodology in bioacoustics, enabling the study of animal-produced sounds [23] in a non-invasive manner that avoids disrupting animal behavior and ecosystem dynamics.

With the advent of the Internet of Things (IoT), audio recorders have become increasingly affordable and energy-efficient, facilitating long-term, in situ monitoring at scale. However, extracting

ecological insights from the resulting massive volumes of continuous data remains an open challenge. Consequently, there is an urgent need for automated, highly efficient processing methodologies that can operate at the edge.

Foundational audio encoders, such as BEATs [30] and AVES [31], have recently demonstrated strong performance in bioacoustic tasks, including bird species classification [168] and whale call detection [169]. Despite their effectiveness, these models are computationally expensive, limiting their deployment on resource-constrained edge devices where near-real-time inference is required. Their size and inference cost may also discourage researchers with limited access to high-end hardware, restricting broader adoption. While naively scaling down these models is possible, doing so often results in substantial performance degradation. We argue that maintaining high accuracy under low-capacity constraints requires designing encoders that learn more meaningful and domain-relevant representations, rather than simply reducing model size.

In this work, we introduce BioME, a resource-efficient self-supervised audio encoder for bioacoustic tasks. BioME is trained via layer-wise knowledge distillation from a Teacher model with strong cross-domain generalization. The Student model is pretrained on a multi-domain corpus comprising speech, environmental sound events, and bioacoustic recordings to further incentivize domain generalization. Furthermore, the central contribution of this work is the integration of modulation-based acoustic features, fused at each Transformer layer through so-called FiLM conditioning [170], which enhances feature disentanglement and representation quality even in low-capacity regimes. Experimental results across multiple bioacoustic benchmarks show that BioME achieves state-of-the-art performance relative to larger models and, in some cases, matches or surpasses its Teacher. Ablation studies further validate the contribution of each architectural and training component.

6.3 Background and Motivation

6.3.1 Bioacoustic Downstream Tasks

The effectiveness of an audio encoder is typically evaluated through downstream benchmarks that fine-tune the model on supervised tasks under standardized evaluation protocols. In bioacoustics, BEANS [171] is one of the widely used benchmarks, comprising 10 bioacoustic tasks and two

general audio tasks (sound event classification and keyword spotting). The bioacoustic tasks span both classification and detection settings, including marine mammal, bat, bird, mosquito, and dog vocalization datasets for classification, as well as detection tasks for birds, mammals, whales, frogs, and Hainan gibbons. Further details on the datasets and evaluation procedures can be found in [171].

In addition, we also adopt the evaluation framework for beehive monitoring established in Chapter 5 and in [121]. This allows us to determine whether a single resource-efficient encoder can generalize effectively beyond standard bioacoustic datasets to specialized tasks such as insect monitoring.

6.3.2 Modulation Spectrum Signal Representation

The modulation spectral signal representation is a signal processing technique designed to quantify the rate-of-change of spectral magnitude components [172]. Signal processing is comprised of two main steps: first, the time-domain signal $x(t)$ is mapped to a spectrotemporal representation $X(t, f)$ (e.g., via short-time Fourier transform). In the second step, a Fast Fourier Transform (FFT) is applied along the time dimension of the spectrotemporal magnitudes $|X(t, f)|$ (i.e., the FFT is applied for each frequency bin), yielding the modulation spectrogram $X(f_{mod}, f)$. This domain characterizes how specific acoustic frequencies f change at modulation frequencies f_{mod}.

This frequency-frequency domain is particularly interesting for bioacoustic data collected "in-the-wild." While environmental noise and artifacts often overlap with bioacoustic signals in both time and acoustic frequency, their rates of change typically differ, making them more disentangled in the modulation domain [173]. This separability enables robust feature extraction even under highly degraded conditions. Furthermore, the representation captures higher-order periodicities, making it highly effective for characterizing animal sounds, which often present strong periodic structures [174, 175], such as bird or whale calls. Lastly, the use of modulation-spectral features for acoustic beehive monitoring has been shown to improve performance for colony strength assessment [176].

To illustrate the advantage of modulation spectral features, we consider the example shown in Figure 6.1. We compute both the spectrogram and the modulation spectrum, averaged across samples, for three hummingbird species with distinct acoustic characteristics: Anna's hummingbird (*annhum*), Broad-tailed hummingbird (*brthum*), and Costa's hummingbird (*coshum*). From the spectrograms (top row), we observe a strong structural similarity across the species. Therefore,

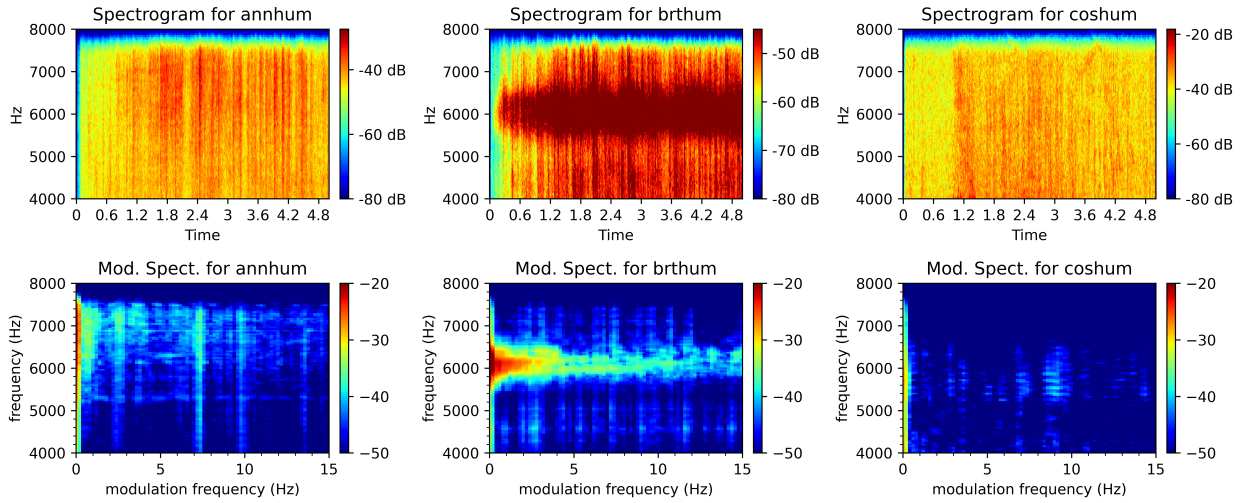


Figure 6.1: Spectrogram (top row) and the modulation spectrum (bottom row) plots, averaged across samples, for three hummingbird species: Anna's hummingbird (*annhum*), Broad-tailed hummingbird (*brthum*), and Costa's hummingbird (*coshum*).

this can make it challenging for classification models relying solely on spectral information to reliably discriminate between them. In contrast, the modulation spectrum (bottom row) better exposes differences by inserting the modulation-frequency dimension. Our hypothesis is that this representation helps disentangle otherwise overlapping spectral structures, thereby providing more discriminative cues and facilitating species separation.

6.4 Proposed Method: The BioME audio encoder

In this section, we describe the main components of the BioME audio encoder. Here, we opted for patch-based features, as in the BEATs encoder. First, given a batch of audio waveforms, we compute mel-spectrograms. Next, the spectrogram is split into non-overlapping patches of size 16×16 and linearly projected. The patch embeddings are then fed to the Transformer encoder layers, which we detail next.

6.4.1 Transformer Encoder

In general, general-purpose audio encoders use a ViT backbone to process patch embeddings. Traditionally, these networks are equipped with convolution-based relative position embeddings. Given recent advances in the natural language processing (NLP) community, especially in Large

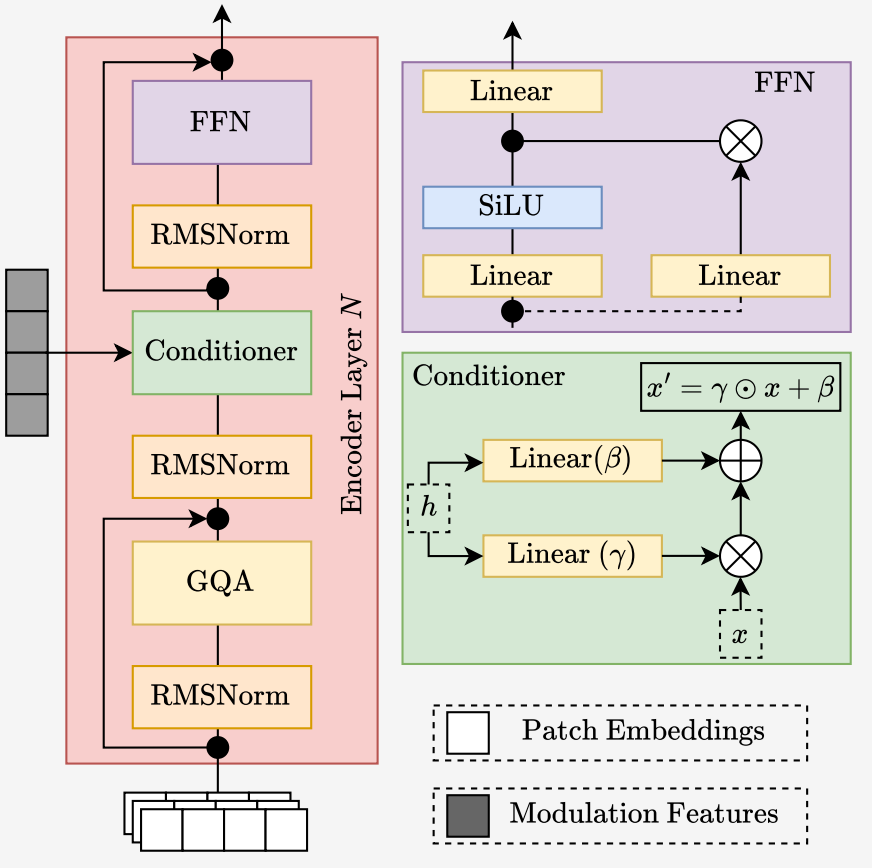


Figure 6.2: Block diagram of a single Transformer layer in the proposed BioME encoder. Patch embeddings are processed alongside side-channel context features, which are integrated at each layer through the Conditioner module implementing FiLM-based conditioning.

Language Models (LLMs), we investigate the use of new components for speech-related tasks. In particular, we adopt some architectural choices from Llama 3.2 [177] for our encoder, as illustrated in Fig. 6.2, and described below.

Grouped Query Attention (GQA) is an attention formulation in which multiple query heads share a smaller set of key-value projections [178]. Originally, it was introduced to reduce memory consumption and accelerate decoding in large language models. However, GQA can also provide practical benefits for compact encoders. By tying key-value representations across groups of heads, GQA reduces the number of projection parameters. Empirically, we find that incorporating GQA into our architecture can benefit edge use cases by lowering VRAM usage for attention computation, reducing latency, and improving training stability in low-capacity regimes.

Rotary Position Embedding (RoPE) [179] is employed to encode positional information directly within the attention mechanism. Unlike the learnable positional embeddings commonly used in ViT-based audio encoders, RoPE requires no additional parameters and integrates seamlessly with multi-head attention, enabling more efficient and scalable sequence modeling while reducing computational overhead.

Furthermore, as depicted in Fig. 6.2, BioME adopts additional architectural components inspired by the LlaMA family of models, including the SiLU activation function and RMSNorm layers. We refer interested readers to the technical report [177] for further details. Lastly, we turn to the integration of modulation-based spectral features via layer-wise conditioning, discussed in the following subsection.

6.4.2 Modulation-features via FiLM

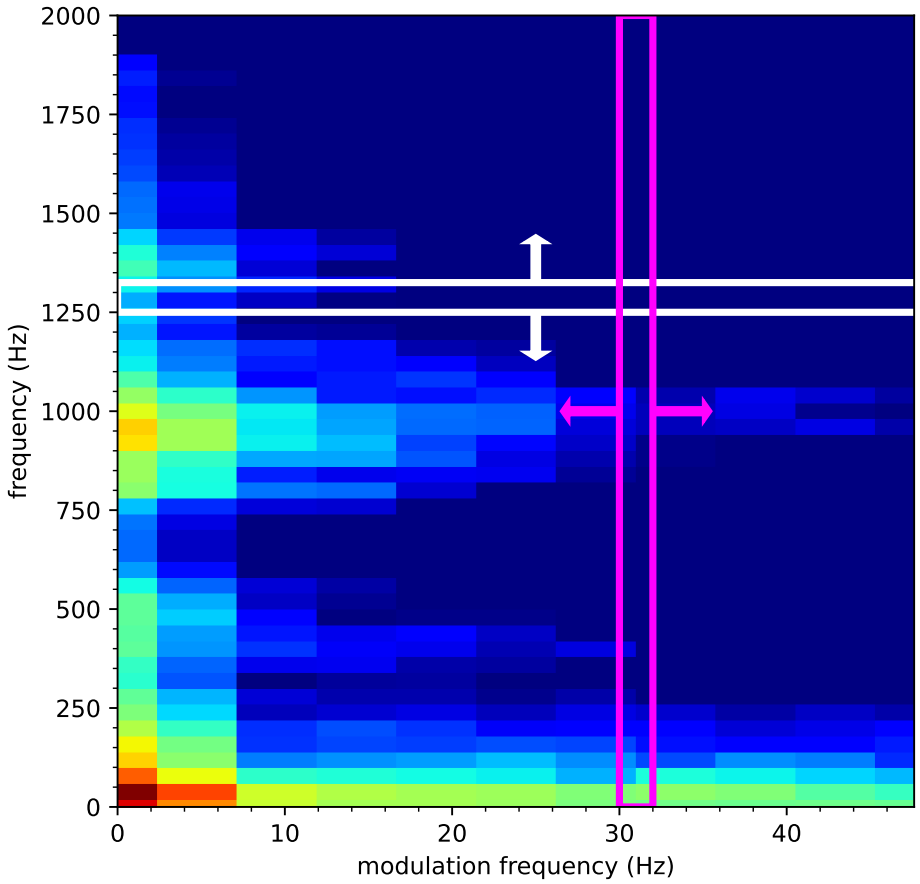


Figure 6.3: Modulation Spectrogram Average Bands (MSAB) computation. For example, a 256×256 modulation spectrogram produces a 512-dimensional MSAB feature vector.

As previously discussed, the modulation spectrogram is a DSP-based tool to capture the dynamic characteristics of frequency components over time. Its output is a two-dimensional modulation spectrogram whose axes correspond to modulation frequency (x-axis) and acoustic frequency (y-axis). Following prior work [176], we compute Modulation Spectrogram Average Bands (MSAB) as a compact summary of this representation. Specifically, we average the modulation spectrogram along both axes, as depicted in Fig. 6.3, and concatenate the resulting vectors that capture band-averaged modulation energies.

The MSAB features are injected into each Transformer layer via a feature-wise linear modulation (FiLM) [170] layer, illustrated in Fig. 6.2 as the 'Conditioner'. Given the (MSAB) context vector h, FiLM computes scale (γ) and shift (β) parameters, which are applied to the contextualized patch embeddings x as $x' = \gamma \odot x + \beta$, where the $\odot$ symbol represents the Hadamard product. Here, γ adjusts the gain of each feature channel, while β applies a bias shift. Unlike cross-attention, FiLM provides a lightweight mechanism for attending intermediate representations with side-channel information. Applied post-attention, it acts as a top-down modulation mechanism [180], effectively incorporating DSP-based inductive biases to enable the model to focus on relevant stimuli and ignore distractions based on modulation features.

6.4.3 Layer-wise Knowledge Distillation

The final component enabling effective training of BioME is knowledge distillation. We adopt BEATs as the Teacher model due to its strong performance on both general-purpose audio tasks and bioacoustic benchmarks. To facilitate layer-wise alignment, our encoder mirrors the Teacher's depth (12 Transformer layers) while exploring different configurations to create a thinner architecture. We perform intermediate-layer distillation at the set of layers $K = \{3, 6, 9, 12\}$ for both the Teacher and the Student, similarly to [74]. When the hidden dimensions differ, a linear projection head maps Student representations to the Teacher's dimensionality.

Our distillation objective is inspired by USAD [74] and DistilHuBERT [45], combining an L1 loss with a cosine similarity term to encourage the Student to match both the magnitude and direction of the Teacher's representations, as described in Equation 2.5.

Table 6.1: Training datasets and configurations used for our distillation experiments with BioME.

			Data Config.	
Dataset	Hours	# Utt.	Core	Bio
AudioSet	4,620.0	1,738,788	✓	✓
FSD50K	80.7	40,966	✓	✓
VGGSound	506.2	182,615	✓	✓
iNatSounds	765.7	137,012		✓
Total (Hours)			5,206.9	5,972.7

6.5 Experimental Setup

6.5.1 Datasets

We employ four datasets under two training configurations, summarized in Table 6.1. The first configuration, *Core*, follows a similar dataset composition used in the AVES encoder [31] and includes AudioSet [181], FSD50K [182], and VGGSound [183]. The second configuration, *Bio*, extends *Core* by incorporating the recently introduced iNatSounds dataset [67], a large-scale bioacoustic collection covering more than 5,500 species. However, approximately 70% of recordings in this dataset are from bird vocalizations. All audio is resampled to 16 kHz for consistency with other encoders.

6.5.2 Hyperparameters and Hardware

We evaluate the scalability of our proposed architecture across three model sizes: an *Edge* variant (6M parameters), a *Small* variant (26M parameters), and a *Base* variant (76M parameters). For the distillation regime, we use the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and a weight decay of 0.01. The learning rate follows a cosine annealing schedule with a linear warm-up: it increases from 10^{-5} to a peak of 10^{-4} over the first 25,000 steps, then decays to zero over the subsequent 75,000 steps (for a total of 100,000 training steps). We use a global batch size of 128 samples. All models are trained on a single Nvidia H100 GPU. At inference time, however, more modest hardware is sufficient. For example, the peak memory to infer 5 minutes of audio (16 kHz sampling rate, FP-32) for the *Edge*, *Small*, and *Base* models is 1.7, 3.6, and 4.3 GB, respectively.

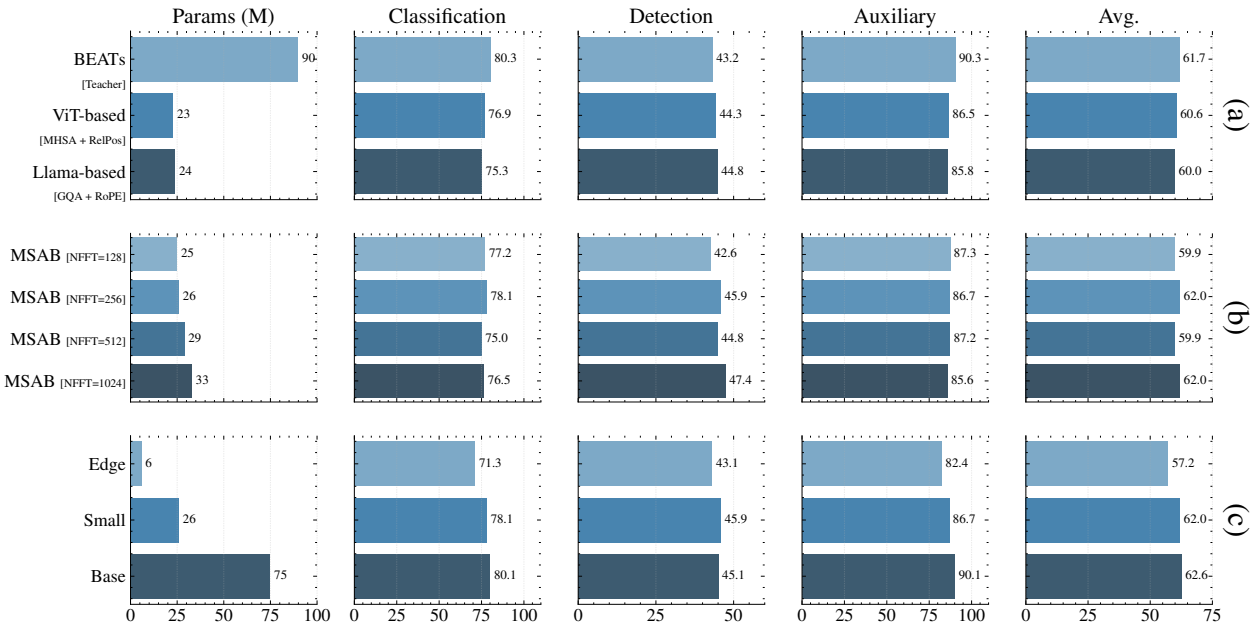


Figure 6.4: Ablation studies for the proposed BioME. We analyze (a) the Student architecture selection, (b) the effect of spectral resolution (NFFT size) on the MSAB features, and (c) model scalability (Edge vs. Small vs. Base). Results are averaged across the classification, detection, and auxiliary tasks of the BEANS benchmark. The 'Avg.' column is the average across all bioacoustic tasks (excluding auxiliary tasks).

6.6 Ablation studies

First, we investigate the effects of our proposed modifications, as summarized in Figure 6.4. The ablation study is divided into three experimental sets: (a) Student Architecture, (b) Spectral Resolution, and (c) Model Scaling. For each configuration, we report five metrics: total model parameters, classification accuracy and detection performance (each averaged across five BEANS tasks), auxiliary task performance, and the overall average across all 10 bioacoustic tasks. Except for the Teacher, all models here are trained using the 'Bio' partition dataset.

Figure 6.4 (a) compares the Teacher model against two Student variants: a "ViT-based" Student (mirroring the Teacher's MHSA + RelPos components but with reduced parameters) and our proposed "Llama-based" architecture. As observed, both Student architectures achieve strong overall performance, remaining competitive with the Teacher despite being approximately four times smaller. We selected the Llama-based architecture for subsequent experiments for two primary reasons: first, its modularity facilitates the integration and testing of side-channel information; second, it offers superior computational efficiency, requiring approximately 5% fewer MMACs/s.

Next, we investigate the impact of modulation spectral features and frequency resolution (Figure 6.4 (b)). Increasing the FFT size reveals a trade-off between classification and detection performance. Larger FFT sizes (e.g., NFFT=512) improved detection tasks, likely by providing the frequency resolution necessary to distinguish events from background noise. However, this comes at the cost of classification accuracy. We found that an intermediate size (NFFT=256) strikes an optimal balance, maintaining high classification accuracy (78.1%) while supporting robust detection capabilities.

Finally, utilizing the optimal spectral features, we evaluate the impact of model scaling (Figure 6.4 (c)) by ablating two more variants: an *Edge* model (6M params) and a *Base* model (75M params). Increasing the model size to the Base configuration yielded the best overall performance, slightly outperforming even the Teacher model (62.6% vs 61.7% average). Conversely, while the Edge model shows a drop in classification performance (from 78.1% to 71.3%), it remains a strong candidate for on-device deployment. The edge model still retains the Teacher's capability while being approximately 12× smaller than the Base model (and 15× smaller than the Teacher), significantly lowering the barrier for edge implementation.

6.7 Results on the BEANS benchmark

Following from our ablation studies, we compare BioME against established baselines on the BEANS benchmark; results are reported in Table 6.2. We report results for our models trained on both the Core and Bio data partitions across all three model sizes. We consistently observe that training on the domain-specific 'Bio' partition yields superior performance compared to the 'Core' partition, confirming the value of in-domain pre-training data. However, we observe only a small performance gap between models trained on the different partitions, particularly for the *Edge* variant. Therefore, this suggests that simply increasing the training data size is not the primary driver of performance for lightweight models; rather, the architectural inductive biases and distillation quality are more critical.

Efficiency and Scalability First, we highlight our results regarding model sizes. When comparing models of similar capacity, our distilled architectures significantly outperform standard audio encoders. Specifically, the BioME *Edge* variant (6M params) achieves a relative improvement of 7.5% over

Table 6.2: Main evaluation results for the BEANS benchmark.

Upstream Information			Bio	Classification					Detection					Avg.
Method	#P (M)	MMACs/s		wtkn	bat	cbi	hbdb	dogs	dcase	enab	hiceas	rfcx	gib	
Supervised Baselines														
rn18	12	4686	—	75.2	44.3	35.7	69.7	66.2	16.1	32.5	28.0	6.4	16.4	39.1
rn50	26	10616	—	79.9	54.8	29.5	69.6	63.3	18.3	28.2	30.4	5.5	21.5	40.1
rn152	60	29807	—	74.3	48.3	33.0	64.5	51.1	15.4	28.0	25.5	6.9	24.8	37.2
vggish	72	864	—	84.7	74.3	44.0	80.8	90.6	33.5	53.5	46.3	14.0	15.0	53.7
rn152p-s	60	29807	—	83.5	60.6	58.3	70.0	79.9	30.0	52.0	32.6	9.7	30.3	50.7
vggish-s	72	864	—	79.4	72.3	44.5	80.1	80.6	32.5	50.9	42.8	14.3	18.6	51.6
Audio SSL Encoders														
BYOL-A [76]	5	143	✗	84.4	75.0	43.3	79.4	87.1	37.2	53.4	53.8	12.6	5.5	53.2
SS-AST [77]	23	10984	✗	91.2	76.6	28.5	81.2	79.1	35.2	51.0	48.8	4.5	30.7	52.7
USAD [74]	24	1353	✗	86.1	77.6	58.3	82.0	97.1	37.2	53.8	63.1	11.3	42.0	60.9
Dasheng [184]	90	2130	✗	77.3	75.5	63.1	69.9	95.7	45.6	43.9	43.8	11.7	41.8	56.8
BEATs [30]	90	4341	✗	89.4	76.8	64.0	80.6	90.7	43.8	49.6	67.4	10.3	44.8	61.7
OpenBEATs [75]	90	4337	✗	87.0	76.9	64.5	84.8	95.7	46.4	12.4	42.3	68.1	49.0	62.7
AVES-bio [31]	89	6635	✓	87.9	74.8	59.8	81.0	95.0	39.2	55.5	62.9	13.0	28.4	59.8
GPTM [185]	89	43797	✓	91.4	77.4	54.3	81.1	94.2	45.4	62.4	65.0	12.9	34.5	61.9
[**Ours**] BioME (Core)	6	227	✓	87.0	70.4	36.5	81.2	86.3	41.2	53.4	60.1	8.0	45.7	57.0
[**Ours**] BioME (Core)	26	1154	✓	89.7	71.6	54.2	81.9	87.1	38.6	59.3	69.9	11.1	50.7	61.4
[**Ours**] BioME (Core)	76	3427	✓	90.9	73.2	60.9	81.6	90.6	39.1	63.3	72.0	7.1	40.6	61.9
[**Ours**] BioME (Bio)	6	227	✓	84.7	68.8	37.2	80.5	85.6	40.0	52.2	66.7	7.9	48.7	57.2
[**Ours**] BioME (Bio)	26	1154	✓	87.0	72.2	55.3	81.2	95.0	43.0	62.1	66.2	11.1	47.3	62.0
[**Ours**] BioME (Bio)	76	3427	✓	91.2	74.3	59.0	81.0	95.0	42.1	62.3	69.3	10.8	40.9	62.6

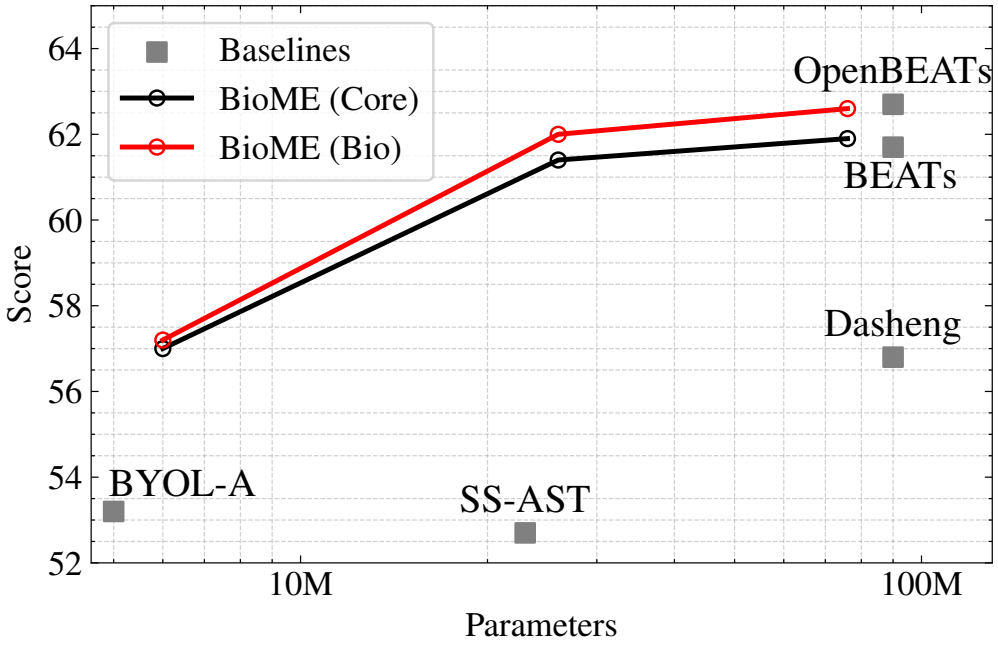


Figure 6.5: Plot showing the relationship between parameter efficiency, scaling properties, and training data partition.

BYOL-A (5M params). Similarly, the BioME *Small* variant (26M params) surpasses SS-AST (23M params) with a substantial gain of 17.6% in overall average score. Figure 6.5 helps to visualize the relationship between model size and performance.

Furthermore, our efficiency gains extend beyond direct size comparisons. The *Edge* model (6M) outperforms significantly larger encoders, including SS-AST (3.8× larger) and Dasheng (15× larger, 90M params), underscoring the effectiveness of the proposed distillation recipe for resource-constrained scenarios.

Teacher-Student Gap At the *Small* (26M) and *Base* scales (76M), we observe that the Student model matches or even slightly exceeds the performance of its Teacher, BEATs (62.0%/62.6% vs. 61.7%). While distilled models are typically bounded by their Teacher's performance, we hypothesize that the inclusion of side-channel information (MSAB features) introduces critical inductive biases that bridge this gap. This phenomenon, of Student models outperforming Teachers by leveraging stronger inductive biases or complementary features, has been similarly observed in prior work [186, 187]. We posit that our proposed recipe, by incorporating these explicit frequency-frequency features, enables the Student to disentangle bioacoustic representations more effectively than the Teacher.

6.8 Acoustic Beehive Monitoring Tasks

Next, we evaluate our models on the specific downstream application of Acoustic Beehive Monitoring (ABM), as detailed in Table 6.3. This benchmark comprises four tasks: Beehive State (BSTS), Beehive Strength (BSTR), Buzzing Identification (BID), and Voice Activity Detection (VAD).

BSTS is a binary classification task designed to detect the presence or absence of a queen bee in a given audio sample. BSTR, a regression task, estimates the colony size; specifically, it predicts the number of hive frames (ranging from 0 to 30) that exhibit at least 80% honey bee coverage. BID is a multi-class classification task that distinguishes between three acoustic events: bee buzzing, cricket chirping, and ambient natural noise. Finally, VAD is a binary task detecting the presence of human speech. Further details on datasets, protocols, and metrics were discussed in Chapter 5 and in one of our papers [121].

Table 6.3: Results for acoustic beehive monitoring tasks. Bold represents the best and underline the second best average results.

Upstream Information		Downstream Tasks				
Method	#P (M) ↓	BSTS ROC-AUC ↑	BSTR MAE ↓	BID ACC ↑	VAD F1-Score ↑	Score
DSP-based Features						
FBanks	0	90.04 (± 2.40)	10.84 (± 0.29)	85.39 (± 9.24)	44.53 (± 8.46)	530
Spectrogram	0	86.75 (± 2.88)	10.95 (± 0.32)	94.49 (± 5.02)	51.10 (± 7.30)	607
MFCC	0	71.42 (± 8.94)	7.22 (± 3.83)	97.95 (± 1.22)	30.28 (± 5.90)	533
General-Purpose Audio Encoders						
BYOL-A	5	63.81 (± 7.61)	3.14 (± 0.21)	98.18 (± 0.91)	63.07 (± 8.83)	733
SS-AST (Tiny)	6	90.24 (± 2.43)	4.44 (± 0.28)	91.66 (± 2.42)	65.13 (± 1.97)	832
SS-AST (Small)	23	92.70 (± 2.05)	4.45 (± 0.16)	96.61 (± 2.03)	72.43 (± 0.96)	910
SS-AST (Base)	89	**93.91** (± 1.44)	4.11 (± 0.22)	74.22 (± 5.95)	72.19 (± 2.32)	706
MSM-MAE	92	89.33 (± 1.81)	4.05 (± 0.13)	97.24 (± 0.81)	73.27 (± 1.16)	863
M2D	85	89.22 (± 3.45)	3.89 (± 0.11)	98.09 (± 0.66)	**74.49** (± 1.37)	883
CAV-MAE	85	83.37 (± 7.72)	3.42 (± 0.22)	93.60 (± 2.50)	69.26 (± 3.31)	797
BEATs	90	75.22 (± 4.30)	**3.05** (± 0.12)	99.44 (± 0.18)	72.82 (± 1.06)	811
Bioacoustic-based Audio Encoders						
AVES	94	90.48 (± 1.80)	3.35 (± 0.16)	**99.46** (± 0.08)	74.42 (± 0.95)	910
BirdAVES (Base)	94	85.32 (± 3.64)	3.39 (± 0.17)	98.22 (± 0.53)	68.59 (± 2.93)	838
BirdAVES (Large)	315	88.98 (± 1.04)	3.56 (± 0.10)	99.02 (± 0.20)	72.72 (± 1.61)	743
[**Ours**] BioME (Bio)	6	92.36 (± 3.03)	4.04 (± 0.26)	97.19 (± 2.62)	68.76 (± 1.16)	**917**
[**Ours**] BioME (Bio)	26	79.62 (± 5.58)	3.63 (± 0.39)	97.96 (± 0.38)	68.03 (± 3.24)	833
[**Ours**] BioME (Bio)	75	70.59 (± 2.69)	3.59 (± 0.11)	98.51 (± 0.42)	73.13 (± 1.88)	770

Performance at the Edge Remarkably, our smallest model, BioME (*Edge*), achieves state-of-the-art results with a top aggregate score of 917. It is important to note that this composite metric explicitly rewards efficiency, with model size accounting for 20% of the final score. However, BioME's success is not driven by size alone; it also outperforms larger baselines like AVES and BEATs on the BSTS task. This task is particularly challenging due to its limited data and strict generalization requirements, as the training and test samples are drawn from different hives. These results highlight the feasibility of deploying specialized, resource-efficient audio encoders directly "in-the-wild" for edge-enabled beehive monitoring.

Inverse Scaling Phenomenon Contrary to the scaling observations in the BEANS benchmark, Table 6.3 reveals an inverse scaling trend on the beehive monitoring tasks. The *Edge* variant (6M) outperforms both the *Small* (26M) and *Base* (75M) variants, achieving the highest overall score (917 vs. 833 and 770, respectively). This trend is primarily driven by the BSTS task, where the Edge

model significantly outperforms its larger counterparts. For the remaining tasks (BSTR, BID, VAD), standard scaling behavior generally applies. We hypothesize that the larger embedding dimension of the Base model ($d = 768$) compared to the Edge model ($d = 192$) results in large probing heads that are prone to overfitting in the binary BSTS classification task. Conversely, the compact feature space of the Edge model acts as a regularizer, forcing the learning of more robust, generalized representations that are particularly effective for this specific classification task.

6.9 Visualizing the impact of MSAB features

Next, to investigate the impact of modulation spectral features on our model, we compare saliency maps with respect to the Mel-filterbank inputs for two configurations: the baseline BioME model (ablated) and the proposed BioME with MSAB features. Figure 6.6 (a) depicts a sample of a Common Dolphin (*Delphinus delphis*) from the Watkins dataset, characterized by significant underwater background noise. An acoustic event is present between the 1 and 2-second marks. While both models correctly attend to the temporal region of the event (Figure 6.6 (b) and (c)), the model utilizing MSAB features (Figure 6.6 (c)) exhibits a distinctly more fine-grained attention pattern. Specifically, it more accurately delineates the harmonic structure and formants of the call compared to the baseline. This observation supports our hypothesis that MSAB features provide important side-channel information, as shown in Figure 6.6 (d), guiding the model to effectively suppress low-frequency background noise and focus on salient bioacoustic structures during prediction.

In addition, to characterize the learned representations, we analyzed the latent space using the Cornell Bird Identification (*cbi*) dataset, which is part of the BEANS benchmark. We evaluated pretrained models in a zero-shot setting to test the hypothesis that modulation-spectral features help with feature disentanglement, particularly in lower-capacity models. We return to the example presented in Figure 6.1 and extract embeddings for the three hummingbird species. Three models are evaluated: BioME (23M), BEATs, and SS-AST. We extract their representation from the last Transformer layer and project it into a two-dimensional space via t-SNE. To quantify separability within this compressed manifold, we fitted a Decision Tree (DT) with a max_depth = 3 and measured the classification accuracy. The results are illustrated in Figure 6.7.

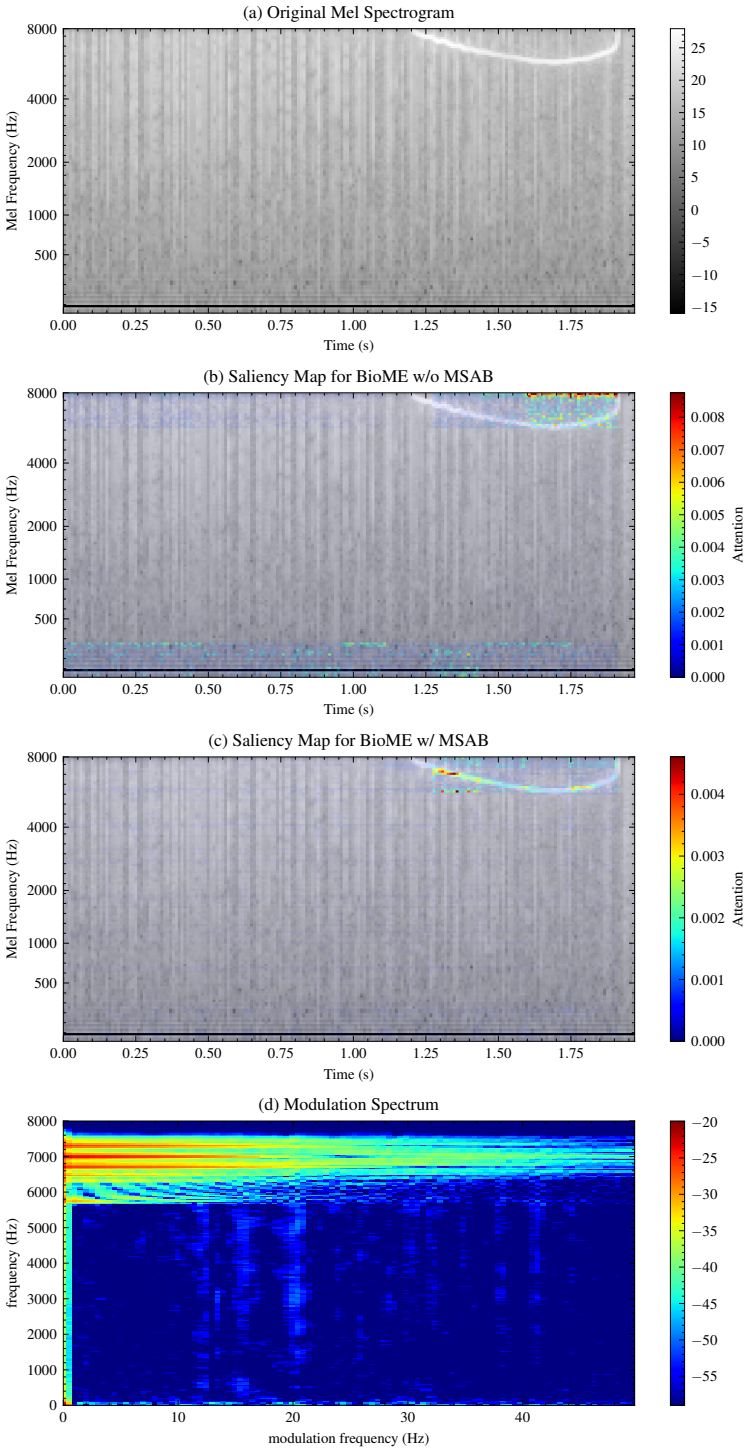


Figure 6.6: Visualization of a Common Dolphin sample (75003041.wav) from the Watkins dataset. The panels display, from top to bottom: (a) the input mel spectrogram; (b) saliency maps from the ablated BioME model (without modulation features); (c) saliency maps from the full BioME model; and (d) the corresponding modulation spectrogram used as side-channel information (prior to MSAB transformation).

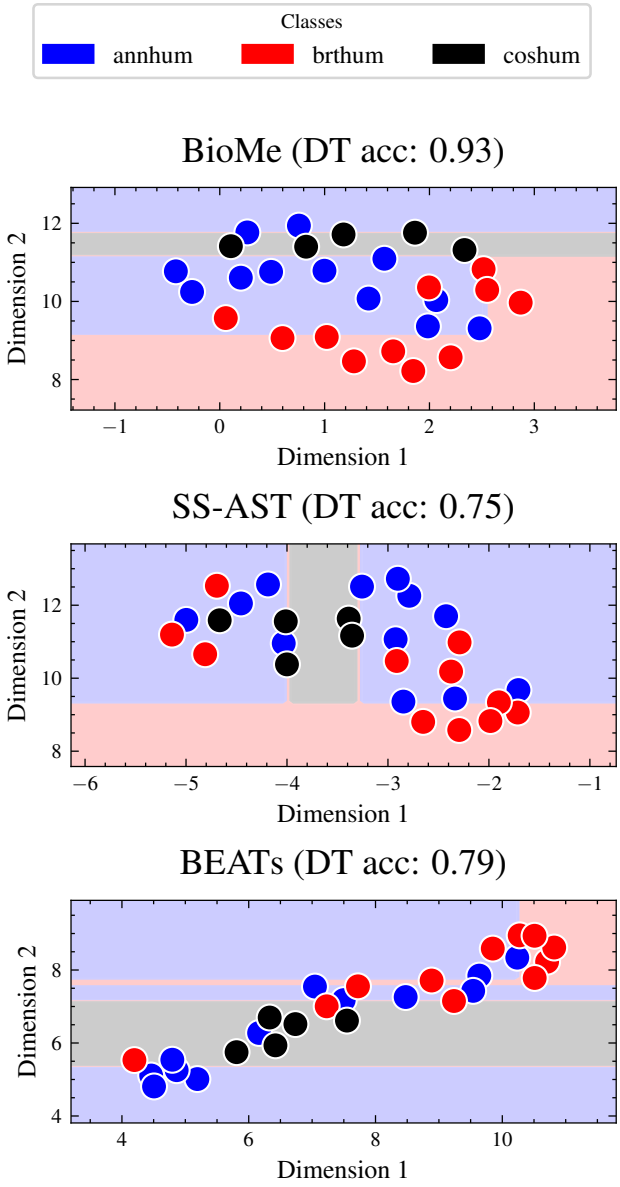


Figure 6.7: t-SNE visualizations of the latent spaces for BioME (Bio partition, 26M), SS-AST, and BEATs using three hummingbird species (Anna's, Broad-tailed, and Costa's) from the Cornell Bird Identification dataset. To quantify disentanglement, a Decision Tree (max depth=3) was fitted to the 2D projections; the resulting decision boundaries and accuracy scores illustrate the latent disentanglement of each model's representation.

Qualitatively (visual inspection) and quantitatively (DT accuracy), we observe that BioME can produce disentangled latent spaces, which helps explain the methodology's overall good performance across different tasks. In this toy example, BEATs takes the second place, followed by SS-AST. While BEATs achieves higher performance on the full 264-class *cbi* task (Table 6.2), where separation occurs in the high-dimensional latent space, this visualization highlights BioME's capacity to maintain distinct class boundaries in a zero-shot manner.

6.10 Conclusions

In this work, we introduced **BioME**, a family of resource-efficient audio encoders designed to enable passive acoustic monitoring on edge devices. By distilling knowledge from the BEATs Teacher into a compact Student, coupled with modulation-spectral features (via FiLM), we demonstrated that strong inductive biases can effectively compensate for reduced model capacity. Our evaluation on the BEANS benchmark confirms this efficiency: our 6M-parameter *Edge* variant outperforms baselines up to 15× its size, while our *Small* (26M) and *Base* (76M) variants surpass the Teacher itself. These results suggest that integrating DSP-inspired features allows Student models to better disentangle complex bioacoustic features than purely data-driven methods. In addition, we demonstrate BioME's practical utility for acoustic beehive monitoring tasks, leading to significant performance gains for small models and achieving state-of-the-art performance, highlighting its versatility for diverse ecological applications.

Chapter 7

Conclusions and Future Research Directions

7.1 Conclusions

In this doctoral thesis, we proposed and evaluated multiple methodologies to bridge the gap between the development of large-scale foundation models and their reliable deployment on resource-constrained edge devices. Specifically, we aimed to mitigate performance degradation caused by "in-the-wild" conditions. In particular, three challenges are investigated in this thesis:

1. **Model Capacity:** Edge devices are inherently constrained by power and storage budgets. While model efficiency techniques (e.g., compression) are well-explored, they typically prioritize maintaining performance on *in-distribution* data (i.e., matching clean test set accuracy). However, compressed models are significantly more vulnerable to perturbations (e.g., environmental noise and adversarial attacks) frequently encountered in real-world deployment.

2. **Distribution Shifts:** The i.i.d. assumption, fundamental to the generalization theory of machine learning algorithms, is often violated in the real-world, where distribution shifts are a natural phenomenon. Empirical evidence presented in this work demonstrates that compressed models are disproportionately sensitive to these shifts compared to their full-capacity counterparts.

3. **Adversarial Examples:** Bad actors may design malicious perturbations to exploit model vulnerabilities, forcing them to behave in unintended ways. These perturbations, known as adversarial attacks, are designed to be imperceptible to human inspection yet sufficient to cause confident mispredictions, posing severe security risks for edge systems.

While significant efforts have been made in the literature to address efficiency, the design of robust, resource-efficient models has not received the same level of attention. This thesis aimed to bridge this gap and facilitate the secure edge deployment of foundation models. A central theme of this work is the incorporation of inductive biases (expert knowledge) into the knowledge distillation process. We demonstrated that this approach yields Student models that are not only compact but also more robust against conditions encountered on edge devices. Our contributions are divided into two main blocks: we first explore speech-processing applications and later bioacoustic applications.

7.1.1 Speech Processing Applications

In verbal communication, speech constitutes the primary mode of human communication, having evolved over generations as an efficient and expressive medium for information exchange. Consequently, it represents a critical and intuitive modality for human and intelligent (computer) systems. However, the speech signal is complex: it carries not only semantic content but also intonation and paralinguistic cues. Furthermore, as speech propagates through the acoustic medium (air), it is inevitably corrupted by environmental factors. Modeling speech signals is therefore a complex challenge, requiring systems capable of disentangling the signal of interest from these confounding factors.

Addressing these challenges, Chapter 3 investigated popular speech foundation models (Wav2Vec 2.0, HuBERT, and WavLM) and their performance under environmental distribution shifts. We show that while these models effectively capture features from raw speech, their performance degrades significantly under unseen perturbations, and that this limitation is even more severe in compressed models due to their reduced capacity. To overcome this, we introduced RobustDistiller, a task-agnostic distillation framework that jointly compresses foundation models and enhances their environmental robustness. By training the Student on noisy inputs while targeting the Teacher's clean representations, the framework leverages two key mechanisms: (i) Feature Denoising Knowledge Distillation, which enforces the learning of noise-invariant representations; and (ii) an auxiliary signal

reconstruction task, which preserves intrinsic signal structure and promotes the disentanglement of speech content from noise.

We demonstrated the versatility of RobustDistiller by successfully adapting it to two distillation mechanisms: layerwise distillation (e.g., DistilHuBERT) and joint distillation-pruning (e.g., DPHuBERT). The robustness of the proposed models, trained with RobustDistiller, was evaluated across clean and noisy conditions on 12 diverse speech-processing tasks from the SUPERB and SUPERB-SG benchmarks. Experimental results indicate that our method frequently outperforms the original Teacher models, particularly in out-of-distribution scenarios, despite operating with approximately 75% fewer parameters. In several instances, the distilled models even surpassed the Teacher's performance on in-domain tasks. Notably, integrating RobustDistiller with the DPWavLM architecture yielded the most consistent performance improvements across all evaluated tasks, confirming its suitability as an efficient and resilient backbone for "in-the-wild" edge applications.

In Chapter 4, we shifted the focus to the security of edge applications, specifically addressing the vulnerability of Keyword Spotting (KS) systems, the initial layer of voice assistants, to adversarial interference. We introduced VIC-KD, a robust knowledge distillation framework designed to compress speech encoders while significantly enhancing their resilience against L_p-norm adversarial attacks. The framework relies on two core mechanisms: (i) the use of Multi-View (MV) inputs to induce the learning of perturbation-invariant features, and (ii) the use of geometric priors via Variance-Invariance-Covariance Regularization (VICReg) on the latent representations of both Teacher and Student models.

Beyond the proposed architecture, this chapter provided critical insights into the nature of robust distillation. First, we demonstrated that Students do not automatically inherit robustness from their Teachers through standard distillation; specific interventions are needed to preserve robustness guarantees. Furthermore, we observed that employing computationally expensive robust Teachers does not necessarily yield superior Students compared to those distilled from standard Teachers via our method. This suggests that for datasets like Google Speech Commands, the computational overhead of pre-training robust Teachers is unjustified when VIC-KD is employed.

We evaluated the framework under diverse scenarios, including clean conditions and a suite of white- and black-box attacks (FGSM, PGD, AutoAttack, and FAKEBOB). Empirically, VIC-KD outperforms state-of-the-art robust distillation methods, improving robust accuracy against

AutoAttack by 12% and 8% over ARD and RSLAD, respectively. Finally, while increasing attack strength (e.g., perturbation budget ϵ or optimization steps) degrades performance across all models, VIC-KD kept a distinct performance margin over other methods. We attribute this stability to the Multi-View strategy, which we hypothesize to maximize the mutual information between views, thereby enforcing better disentanglement of speech features from adversarial noise.

7.1.2 Bioacoustic Applications

Next, we turn our focus towards bioacoustic applications. In Chapter 5, we focus on IoT-enabled acoustic beehive monitoring. We present a comprehensive benchmark of general-purpose and bioacoustic audio encoders across four tasks: beehive state detection, beehive strength assessment, buzzing identification, and beekeeper voice activity detection. We evaluate fourteen models and release our code [1] to enable future research on advanced and scalable honeybee colony monitoring.

From our analysis, we obtained key insights into the deployment of SSL models for ecological monitoring. First, we observed that while models trained on diverse audio corpora provide a strong initialization, and domain-specific adaptation (e.g., continual pretraining with bee buzzing data) can significantly enhance downstream performance. Second, we identified a crucial trade-off: while most foundation models excel in in-distribution scenarios, out-of-distribution generalization remains a challenge, with handcrafted features occasionally serving as competitive baselines for complex state classification tasks.

From a resource-efficiency perspective, our findings highlight the viability of compact edge architectures. Notably, smaller models like SS-AST (Small) achieved competitive accuracy despite their reduced footprint, confirming their suitability for low-power IoT devices. These results set the stage for the final phase of the thesis, motivating the exploration of advanced distillation techniques that combine efficiency and expert knowledge to better design good Student models in Chapter 6.

Finally, in Chapter 6, we propose BioME (Bioacoustic Modulation-aware Encoder), a family of resource-efficient architectures designed specifically to enable passive acoustic monitoring on edge devices. BioME is trained via knowledge distillation from a high-capacity BEATs Teacher into a compact Student. To improve efficiency, we incorporated architectural optimizations from

[1] `https://github.com/MuSAELab/hunee-b`

the LLM literature, such as Grouped Query Attention (GQA) and Rotary Positional Embeddings (RoPE). Furthermore, we introduced a novel inductive bias into the distillation process: the fusion of Modulation Spectrum features as side-channel information at each Transformer layer. This design leverages the property that bioacoustic signals and environmental noise, while often overlapping in time and frequency, are typically separable in the modulation domain based on their rates of change, and enables us to exploit the periodic nature of animal vocalizations.

We validated BioME across the BEANS and our specific beehive monitoring benchmark, demonstrating that it consistently breaks the traditional trade-off between model size and accuracy. When compared to baselines of similar capacity, BioME achieves substantial margins; specifically, the BioME Edge (6M) variant outperforms BYOL-A (5M) by 7.5%, while BioME Small (26M) surpasses SS-AST (23M) by 17.6%. Remarkably, our distilled models outperformed substantially larger foundation models (e.g., AVES and BEATs) on challenging out-of-distribution tasks, such as the beehive state classification, where training and test distributions come from different hives.

Lastly, we analysed in more depth our hypothesis that modulation-aware conditioning promotes better representation learning. Saliency map visualizations revealed that BioME shows more fine-grained attention to harmonic structures and formants than ablated baselines. Furthermore, analysis of the latent space in a zero-shot hummingbird dataset demonstrated that BioME produces significantly better feature disentanglement than competitive encoders. These results confirm that integrating DSP-based priors via modulation features is a key enabler for deploying robust, high-performance bioacoustic monitors in the wild.

7.2 Future Research Directions

In the following sections, we delineate potential avenues for future research building upon the contributions of this thesis. We also address specific limitations of the proposed frameworks and outline directions to enhance the deployment of edge-native audio intelligence.

7.2.1 The self-attention bottleneck

While our work demonstrated significant reductions in model footprint, achieving compression rates of approximately 4× with RobustDistiller and 15× with BioME, parameter count is not the sole determinant of inference efficiency. A critical remaining bottleneck is the quadratic time and memory complexity of the standard self-attention mechanism. This limitation is particularly severe in the bioacoustic domain. Although our evaluation focused on 16 kHz audio, effective monitoring of many species (e.g., bats, cetaceans) requires significantly higher sampling rates (ultrasonic ranges), which proportionally increase the sequence length and cause the computational cost to explode.

Linear-complexity attention mechanisms have started to demonstrate promising results in automatic speech recognition [188] and speech synthesis [189]. An interesting avenue for future research is the exploration of cross-architectural distillation, where knowledge is transferred from a high-capacity Teacher with quadratic attention to a Student utilizing linear attention (e.g., state-space models or linear transformers). This approach would be especially valuable for bioacoustic monitoring, where analyzing long-duration recordings without losing context via windowing remains a non-trivial challenge.

7.2.2 Test-time Adaptation for Edge Applications

In this thesis, our robustness strategies have largely relied on anticipating what specific degradation categories would be present in the test signal (e.g., environmental noise, reverberation, adversarial perturbations). While our evaluation protocols tested the generalization to unseen instances of these factors, real-world deployment often introduces entirely unforeseen anomalies, ranging from transmission artifacts (e.g., packet loss concealment) to complex synthetic spoofing attacks (e.g., deepfakes). Since exhaustively modeling every distortion during training is intractable, this opens a new research venue for our work.

Two potential directions are Domain Generalization (DG) [190], which aims to learn representations without making assumptions about test-time shifts, and Test-Time Adaptation (TTA). TTA seeks to adapt a pre-trained model to the target distribution using unlabeled data streams directly at inference time, effectively calibrating the model before a prediction is made. While TTA has been explored for speech recognition [191] and enhancement [192], its application to resource-constrained

edge devices and audio foundation models remains an open challenge. Standard TTA methods often require backpropagation during inference, which might be prohibitively expensive in small devices. Future research should therefore focus on developing efficient TTA algorithms that update only lightweight normalization statistics or bias terms, allowing bioacoustic monitors to dynamically adapt to new acoustic environments (e.g., a new apiary or rainforest) without the need for cloud-based retraining or heavy on-device computation.

7.2.3 Leveraging synthetic data for audio applications

In Chapters 5 and 6, we discussed how labelled ecological acoustic data might be difficult to acquire, often requiring specialized professionals to manually inspect and categorize vast amounts of data. An interesting research direction, already being explored in the speech domain [193, 194], is the use of generative modeling approaches (e.g., diffusion models) to create realistic synthetic data. For example, while approaches have been recently explored in the bioacoustic domain [195, 196], there is still significant room for improvement. In [197], for instance, the authors explore a WaveGAN-variant model to synthesize honeybee acoustic events (e.g., queen presence vs. absence), successfully using the synthetic data to augment the training set and improve the robustness of downstream classifiers.

A critical challenge in creating synthetic data is to investigate controllable generation techniques, in which models are conditioned not only on class labels but also on physical parameters (e.g., hive size, temperature, or environment). This would enable the targeted synthesis of important scenarios, such as rare distress signals or specific environmental noise conditions, that are unlikely to be captured during standard data collection, thereby directly addressing the out-of-distribution robustness issues identified in this thesis.

Bibliography

[1] Yi Zhu, Mahsa Abdollahi, Ségolène Maucourt, Nico Coallier, Heitor R Guimarães, Pierre Giovenazzo, and Tiago H Falk. MSPB: a longitudinal multi-sensor dataset with phenotypic trait measurements from honey bees. *Scientific Data*, 11(1):860, 2024.

[2] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023.

[3] Krishna C Puvvada, Piotr Żelasko, He Huang, Oleksii Hrinchuk, Nithin Rao Koluguri, Kunal Dhawan, Somshubra Majumdar, Elena Rastorgueva, Zhehuai Chen, Vitaly Lavrukhin, et al. Less is more: Accurate speech recognition & translation without web-scale data. In *Proc. Interspeech 2024*, pages 3964–3968, 2024.

[4] Youzhi Tu, Weiwei Lin, and Man-Wai Mak. A survey on text-dependent and text-independent speaker verification. *IEEE Access*, 10:99038–99049, 2022.

[5] Jörg Müller, Oliver Mitesser, H Martin Schaefer, Sebastian Seibold, Annika Busse, Peter Kriegel, Dominik Rabl, Rudy Gelis, Alejandro Arteaga, Juan Freile, et al. Soundscapes and deep learning enable tracking biodiversity recovery in tropical forests. *Nature communications*, 14(1):6191, 2023.

[6] Carlos Alberto Vargas Rivera. *Hearing the Forest: Machine Learning for Biodiversity Monitoring Using Soundscapes*. PhD thesis, Technische Universität Wien, 2025.

[7] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: a review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[8] Jessica AF Thompson, Yoshua Bengio, Elia Formisano, and Marc Schönwiesner. How can deep learning advance computational modeling of sensory information processing? *arXiv preprint arXiv:1810.08651*, 2018.

[9] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012.

[10] Purvi Agrawal and Sriram Ganapathy. Interpretable representation learning for speech and audio signals based on relevance weighting. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2823–2836, 2020.

[11] Rishi Bommasani. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[13] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, et al. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 29:3451–3460, oct 2021.

[14] Li Wang, Jiaqi Li, Yuhao Luo, Jiahao Zheng, Lei Wang, Hao Li, Ke Xu, Chengfang Fang, Jie Shi, and Zhizheng Wu. Advsv: An over-the-air adversarial attack dataset for speaker verification. In *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4555–4559. IEEE, 2024.

[15] Lawrence R Rabiner and Ronald W Schafer. Introduction to digital speech processing. *Foundations and Trends in Signal Processing*, 1(1-2):1–194, 2007.

[16] W Tecumseh Fitch. *The evolution of language*. Cambridge University Press, 2010.

[17] Douglas O'Shaughnessy. Review of analysis methods for speech applications. *Speech Communication*, 151:64–75, 2023.

[18] Yi Zhu and Tiago H. Falk. Fusion of modulation spectral and spectral features with symptom metadata for improved speech-based covid-19 detection. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8997–9001, 2022.

[19] Tawfiq Ammari, Jofish Kaye, Janice Y Tsai, and Frank Bentley. Music, search, and iot: How people (really) use voice assistants. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 26(3):1–28, 2019.

[20] Stuart HM Butchart, Matt Walpole, Ben Collen, Arco Van Strien, Jörn PW Scharlemann, Rosamunde EA Almond, Jonathan EM Baillie, Bastian Bomhard, Claire Brown, John Bruno, et al. Global biodiversity: indicators of recent declines. *Science*, 328(5982):1164–1168, 2010.

[21] Bradley J Cardinale, J Emmett Duffy, Andrew Gonzalez, David U Hooper, Charles Perrings, Patrick Venail, Anita Narwani, Georgina M Mace, David Tilman, David A Wardle, et al. Biodiversity loss and its impact on humanity. *Nature*, 486(7401):59–67, 2012.

[22] Fazal Ullah, Saddam Saqib, and You-Cai Xiong. Integrating artificial intelligence in biodiversity conservation: bridging classical and modern approaches. *Biodiversity and Conservation*, 34(1):45–65, 2025.

[23] Daniella Teixeira, Martine Maron, and Berndt J van Rensburg. Bioacoustic monitoring of animal vocal behavior for conservation. *Conservation Science and Practice*, 1(8):e72, 2019.

[24] Daniella Teixeira, Paul Roe, Berndt J van Rensburg, Simon Linke, Paul G McDonald, David Tucker, and Susan Fuller. Effective ecological monitoring using passive acoustic sensors: Recommendations for conservation practitioners. *Conservation Science and Practice*, 6(6):e13132, 2024.

[25] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[27] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems*, 33:12449–12460, 2020.

[28] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[29] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[30] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, Wanxiang Che, Xiangzhan Yu, and Furu Wei. BEATs: Audio pre-training with acoustic tokenizers. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 5178–5193. PMLR, 23–29 Jul 2023.

[31] Masato Hagiwara. Aves: Animal vocalization encoder based on self-supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[32] Shu-wen Yang, Po-Han Chi, et al. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*, 2021.

[33] Hsiang-Sheng Tsai, Heng-Jui Chang, Wen-Chin Huang, Zili Huang, Kushal Lakhotia, Shu-wen Yang, Shuyan Dong, Andy Liu, Cheng-I Lai, Jiatong Shi, Xuankai Chang, Phil Hall, Hsuan-Jui Chen, Shang-Wen Li, Shinji Watanabe, Abdelrahman Mohamed, and Hung-yi Lee. SUPERB-SG: Enhanced speech processing universal PERformance benchmark for semantic and generative capabilities. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8479–8492, Dublin, Ireland, May 2022.

[34] Masato Hagiwara, Benjamin Hoffman, Jen-Yu Liu, Maddie Cusimano, Felix Effenberger, and Katie Zacarian. Beans: The benchmark of animal sounds. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[35] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. *Advances in neural information processing systems*, 28, 2015.

[36] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Lingqiao Liu, and Ian Reid. Towards effective low-bitwidth convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7920–7928, 2018.

[37] Haoning Xu, Zhaoqing Li, Zengrui Jin, Huimeng Wang, Youjun Chen, Guinan Li, Mengzhe Geng, Shujie Hu, Jiajun Deng, and Xunying Liu. Effective and efficient mixed precision quantization of speech foundation models. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[38] Yueqian Lin, Yuzhe Fu, Jingyang Zhang, Yudong Liu, Jianyi Zhang, Jingwei Sun, Hai Helen Li, and Yiran Chen. Speechprune: Context-aware token pruning for speech information retrieval. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2025.

[39] Bei Liu, Tianteng Gu, Haoyu Wang, and Yanmin Qian. Mixpq: Joint pruning and quantization for speech and language foundation models compression. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

[40] Mouaad Oujabour, Leila Ben Letaifa, Jean-François Dollinger, and Jean-Luc Rouas. Adaptive compression of supervised and self-supervised models for green speech recognition. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[41] Marva Touheed, Urooj Zubair, Dilshad Sabir, Ali Hassan, Muhammad Fasih Uddin Butt, Farhan Riaz, Wadood Abdul, and Rashid Ayub. Applications of pruning methods in natural language processing. *IEEE Access*, 12:89418–89438, 2024.

[42] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[43] Gustavo Aguilar, Yuan Ling, Yu Zhang, Benjamin Yao, Xing Fan, and Chenlei Guo. Knowledge distillation from internal representations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7350–7357, 2020.

[44] Siqi Sun, Zhe Gan, Yuwei Fang, Yu Cheng, Shuohang Wang, and Jingjing Liu. Contrastive distillation on intermediate representations for language model compression. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 498–508, 2020.

[45] Heng-Jui Chang, Shu-wen Yang, and Hung-yi Lee. Distilhubert: Speech representation learning by layer-wise distillation of hidden-unit bert. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7087–7091. IEEE, 2022.

[46] Rui Wang, Qibing Bai, Junyi Ao, Long Zhou, Zhixiang Xiong, Zhihua Wei, Yu Zhang, Tom Ko, and Haizhou Li. Lighthubert: Lightweight and configurable speech representation learning with once-for-all hidden-unit bert. *arXiv preprint arXiv:2203.15610*, 2022.

[47] Yifan Peng, Yui Sudo, Shakeel Muhammad, and Shinji Watanabe. DPHuBERT: Joint Distillation and Pruning of Self-Supervised Speech Models. In *Proc. INTERSPEECH 2023*, pages 62–66, 2023.

[48] Heitor R. Guimarães, Arthur Pimentel, Anderson R. Avila, Mehdi Rezagholizadeh, Boxing Chen, and Tiago H. Falk. Robustdistiller: Compressing universal speech representations for enhanced environment robustness. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[49] Wei-Ning Hsu, Anuroop Sriram, Alexei Baevski, Tatiana Likhomanenko, Qiantong Xu, Vineel Pratap, Jacob Kahn, Ann Lee, Ronan Collobert, Gabriel Synnaeve, et al. Robust wav2vec 2.0: Analyzing domain shift in self-supervised pre-training. *arXiv preprint arXiv:2104.01027*, 2021.

[50] Kuan Po Huang, Yu-Kuan Fu, Yu Zhang, and Hung yi Lee. Improving Distortion Robustness of Self-supervised Speech Processing Tasks with Domain Adaptation. In *Proc. Interspeech 2022*, pages 2193–2197, 2022.

[51] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

[52] A. Madry, A. Makelov, L. Schmidt, et al. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[53] Haibin Wu, Bo Zheng, Xu Li, Xixin Wu, Hung-yi Lee, and Helen Meng. Characterizing the adversarial vulnerability of speech self-supervised learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3164–3168. IEEE, 2022.

[54] M. Goldblum, L. Fowl, S. Feizi, et al. Adversarially robust distillation. In *Proc. of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3996–4003, 2020.

[55] B. Zi, S. Zhao, X. Ma, et al. Revisiting adversarial robustness distillation: Robust soft labels make student better. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16443–16452, 2021.

[56] Heming Wang, Yao Qian, Xiaofei Wang, Yiming Wang, Chengyi Wang, Shujie Liu, Takuya Yoshioka, Jinyu Li, and DeLiang Wang. Improving noise robustness of contrastive speech representation learning with speech reconstruction. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6062–6066. IEEE, 2022.

[57] Heitor R. Guimarães, Yi Zhu, Orson Mengara, Anderson R. Avila, and Tiago H. Falk. Assessing the vulnerability of self-supervised speech representations for keyword spotting under white-box adversarial attacks. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1672–1677, 2023.

[58] A. Bardes, J. Ponce, and Y. LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

[59] P. Warden. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209*, 2018.

[60] M. Ravanelli, T. Parcollet, P. Plantinga, et al. SpeechBrain: A general-purpose speech toolkit, 2021. arXiv:2106.04624.

[61] S. Choi, S. Seo, B. Shin, et al. Temporal Convolution for Real-Time Keyword Spotting on Mobile Devices. In *Proc. Interspeech 2019*, pages 3372–3376, 2019.

[62] D. Snyder, D. Garcia-Romero, G. Sell, et al. X-vectors: Robust dnn embeddings for speaker recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5329–5333, 2018.

[63] IZSLT CAAS. Good beekeeping practices for sustainable apiculture. `https://www.fao.org/family-farming/detail/en/c/1442505/`, 2021. Accessed: 2026-03-01.

[64] Mahsa Abdollahi, Pierre Giovenazzo, and Tiago H Falk. Automated beehive acoustics monitoring: A comprehensive review of the literature and recommendations for future work. *Applied Sciences*, 12(8):3920, 2022.

[65] Tony Zhang, Szymon Zmyslony, Sergei Nozdrenkov, Matthew Smith, and Brandon Hopkins. Semi-supervised audio representation learning for modeling beehive strengths. *arXiv preprint arXiv:2105.10536*, 2021.

[66] Mahsa Abdollahi, Evan Henry, Pierre Giovenazzo, and Tiago H Falk. The importance of context awareness in acoustics-based automated beehive monitoring. *Applied Sciences*, 13(1):195, 2022.

[67] Mustafa Chasmai, Alexander Shepard, Subhransu Maji, and Grant Van Horn. The inaturalist sounds dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[68] Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[69] Abdelrahman Mohamed, Dmytro Okhonko, and Luke Zettlemoyer. Transformers with convolutional context for asr. *arXiv preprint arXiv:1904.11660*, 2019.

[70] Felix Wu, Angela Fan, Alexei Baevski, Yann Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. In *International Conference on Learning Representations*, 2019.

[71] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017.

[72] Douglas O'shaughnessy. *Speech communications: Human and machine (IEEE)*. Universities press, 1987.

[73] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, with Language Models*. 3rd edition, 2026. Online manuscript released January 6, 2026.

[74] Heng-Jui Chang, Saurabhchand Bhati, James Glass, and Alexander H Liu. Usad: Universal speech and audio representation via distillation. *arXiv preprint arXiv:2506.18843*, pages 1–8, 2025.

[75] Shikhar Bharadwaj, Samuele Cornell, Kwanghee Choi, Satoru Fukayama, Hye-Jin Shim, Soham Deshmukh, and Shinji Watanabe. Openbeats: A fully open-source general-purpose audio encoder. In *2025 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 1–5, 2025.

[76] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. BYOL for Audio: Exploring pre-trained general-purpose audio representations. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:137–151, 2023.

[77] Yuan Gong, Cheng-I Lai, Yu-An Chung, and James Glass. Ssast: Self-supervised audio spectrogram transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10699–10709, 2022.

[78] Salah Zaiem, Youcef Kemiche, Titouan Parcollet, Slim Essid, and Mirco Ravanelli. Speech self-supervised representation benchmarking: Are we doing it right? In *INTERSPEECH 2023*, pages 2873–2877. ISCA, 2023.

[79] Salah Zaiem, Youcef Kemiche, Titouan Parcollet, Slim Essid, and Mirco Ravanelli. Speech self-supervised representations benchmarking: A case for larger probing heads. *Computer Speech & Language*, 89:101695, 2025.

[80] Tzu-hsun Feng, Annie Dong, Ching-Feng Yeh, Shu-wen Yang, Tzu-Quan Lin, Jiatong Shi, Kai-Wei Chang, Zili Huang, Haibin Wu, Xuankai Chang, et al. Superb@ slt 2022: Challenge on generalization and efficiency of self-supervised speech representation learning. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 1096–1103. IEEE, 2023.

[81] Lu Wei, Zhong Ma, Chaojie Yang, and Qin Yao. Advances in the neural network quantization: A comprehensive review. *Applied Sciences*, 14(17):7445, 2024.

[82] Hongrong Cheng, Miao Zhang, and Javen Qinfeng Shi. A survey on deep neural network pruning: Taxonomy, comparison, analysis, and recommendations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[83] Mengzhou Xia, Zexuan Zhong, and Danqi Chen. Structured pruning learns compact and accurate models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1513–1528, 2022.

[84] Ziheng Wang, Jeremy Wohlwend, and Tao Lei. Structured pruning of large language models. In *Proceedings of the 2020 conference on empirical methods in natural language processing (emnlp)*, pages 6151–6162, 2020.

[85] Yeonghyeon Lee, Kangwook Jang, Jahyun Goo, Youngmoon Jung, and Hoi Rin Kim. FitHuBERT: Going Thinner and Deeper for Knowledge Distillation of Speech Self-Supervised Models. In *Proc. Interspeech 2022*, pages 3588–3592, 2022.

[86] Takanori Ashihara, Takafumi Moriya, Kohei Matsuura, and Tomohiro Tanaka. Deep versus Wide: An Analysis of Student Architectures for Task-Agnostic Knowledge Distillation of Self-Supervised Speech Models. In *Proc. Interspeech 2022*, pages 411–415, 2022.

[87] Christos Louizos, Max Welling, and Diederik P. Kingma. Learning sparse neural networks through l_0 regularization. In *International Conference on Learning Representations*, 2018.

[88] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[89] Jacob Kahn, Morgane Rivière, et al. Libri-light: A benchmark for ASR with limited or no supervision. In *Proc. IEEE ICASSP*, pages 7669–7673, 2020.

[90] K.P. Huang, Y.K. Fu, T.Y. Hsu, et al. Improving generalizability of distilled self-supervised speech processing models under distorted settings. In *2022 IEEE Spoken Language Technology Workshop*, pages 1112–1119, 2023.

[91] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[92] V. Vapnik. Principles of risk minimization for learning theory. In J. Moody, S. Hanson, and R.P. Lippmann, editors, *Advances in Neural Information Processing Systems*, volume 4. Morgan-Kaufmann, 1991.

[93] Zico Kolter and Aleksander Madry. Adversarial robustness - theory and practice. NeurIPS 2018 Tutorial, 2018. Accessed: 2024-05-20.

[94] Heitor R. Guimarães, Yi Zhu, Orson Mengara, Anderson R. Avila, and Tiago H. Falk. Assessing the vulnerability of self-supervised speech representations for keyword spotting under white-box adversarial attacks. In *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1672–1677, 2023.

[95] Tiago H. Falk and Wai-Yip Chan. Temporal dynamics for blind measurement of room acoustical parameters. *IEEE Transactions on Instrumentation and Measurement*, 59(4):978–989, 2010.

[96] Heitor Guimarães, Arthur Pimentel, Anderson Avila, Mehdi Rezagholizadeh, and Tiago H. Falk. An exploration into the performance of unsupervised cross-task speech representations for "in the wild" edge applications, 2023.

[97] Heitor Guimarães, Arthur Pimentel, Anderson R Avila, Mehdi Rezagholizadeh, and Tiago H Falk. Improving the robustness of distilhubert to unseen noisy conditions via data augmentation, curriculum learning, and multi-task enhancement. *NeurIPS 2022 Efficient Natural Language and Speech Processing Workshop*, 2022.

[98] Michael L. Seltzer, Dong Yu, and Yongqiang Wang. An investigation of deep neural networks for noise robust speech recognition. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7398–7402, 2013.

[99] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 2, pages 749–752 vol.2, 2001.

[100] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:3451–3460, 2021.

[101] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48, 2009.

[102] David Snyder, Guoguo Chen, and Daniel Povey. MUSAN: A Music, Speech, and Noise Corpus, 2015. arXiv:1510.08484v1.

[103] J. Salamon, C. Jacoby, and J. P. Bello. A dataset and taxonomy for urban sound research. In *22nd ACM International Conference on Multimedia (ACM-MM'14)*, pages 1041–1044, Orlando, FL, USA, Nov. 2014.

[104] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur. A study on data augmentation of reverberant speech for robust speech recognition. In *Proc. IEEE ICASSP*, pages 5220–5224, 2017.

[105] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. A multi-device dataset for urban acoustic scene classification. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018)*, pages 9–13, November 2018.

[106] Arthur Pimentel, Heitor R. Guimarães, Anderson Avila, and Tiago H. Falk. Environment-aware knowledge distillation for improved resource-constrained edge speech recognition. *Applied Sciences*, 13(23), 2023.

[107] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, 2011.

[108] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey. Sdr–half-baked or well done? In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630. IEEE, 2019.

[109] Tiago H. Falk and Wai-Yip Chan. Modulation spectral features for robust far-field speaker identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1):90–100, 2010.

[110] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *INTERSPEECH*, 2017.

[111] Heitor R. Guimarães, Arthur Pimentel, Anderson Avila, and Tiago H. Falk. Vic-kd: Variance-invariance-covariance knowledge distillation to make keyword spotting more robust against adversarial attacks. In *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12196–12200, 2024.

[112] C. Zhao, Z. Li, H. Ding, et al. Utio: Universal, targeted, imperceptible and over-the-air audio adversarial example. In *2022 IEEE 28th International Conference on Parallel and Distributed Systems*, pages 346–353, 2023.

[113] H. Zhang, Y. Yu, J. Jiao, E. Xing, et al. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.

[114] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

[115] J. Zbontar, Li Jing, I. Misra, et al. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

[116] F. Croce and M. Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[117] F. Croce and M. Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, pages 2196–2205. PMLR, 2020.

[118] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[119] Guangke Chen, Sen Chenb, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. Who is real bob? adversarial attacks on speaker recognition systems. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 694–711. IEEE, 2021.

[120] Heitor R. Guimarães, Mahsa Abdollahi, Yi Zhu, Ségolène Maucourt, Nico Coallier, Pierre Giovenazzo, and Tiago H. Falk. Adapting self-supervised features for background speech detection in beehive audio recordings. In *2023 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 663–667, 2023.

[121] Heitor R. Guimarães, Mahsa Abdollahi, Yi Zhu, Ségolène Maucourt, Nicolas Coallier, Pierre Giovenazzo, and Tiago H. Falk. Benchmarking self-supervised audio representations for iot-enabled acoustic beehive monitoring. *IEEE Internet of Things Journal*, 12(21):45000–45010, 2025.

[122] Nirmal Anand, V Bharath Raj, MS Ullas, and Ashutosh Srivastava. Swarm detection and beehive monitoring system using auditory and microclimatic analysis. In *2018 3rd International Conference on Circuits, Control, Communication and Computing (I4C)*, pages 1–4. IEEE, 2018.

[123] Bianca Sousa Soares, Jederson Sousa Luz, Valderlândia Francisca de Macêdo, et al. Mfcc-based descriptor for bee queen presence detection. *Expert Systems with Applications*, 201:117104, 2022.

[124] Yangguang Zhao, Guoqing Deng, Long Zhang, Nayan Di, Xueli Jiang, and Zhigang Li. Based investigate of beehive sound to detect air pollutants by machine learning. *Ecological Informatics*, 61:101246, 2021.

[125] Alison Pereira Ribeiro, Nádia Felix Felipe da Silva, Fernanda Neiva Mesquita, Priscila de Cássia Souza Araújo, Thierson Couto Rosa, and José Neiva Mesquita-Neto. Machine learning approach for automatic recognition of tomato-pollinating bees based on their buzzing-sounds. *PLoS computational biology*, 17(9):e1009426, 2021.

[126] Ilyass Moummad, Nicolas Farrugia, and Romain Serizel. Regularized contrastive pre-training for few-shot bioacoustic sound detection. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1436–1440, 2024.

[127] Hicham Bellafkir, Markus Vogelbacher, Daniel Schneider, et al. Bird species recognition in soundscapes with self-supervised pre-training. In Akram Bennour, Ahmed Bouridane, and Lotfi Chaari, editors, *Intelligent Systems and Pattern Recognition*, pages 60–74, Cham, 2024. Springer Nature Switzerland.

[128] Peter C Bermant, Leandra Brickson, and Alexander J Titus. Bioacoustic event detection with self-supervised contrastive learning. *bioRxiv*, pages 2022–10, 2022.

[129] Jort F. Gemmeke, Daniel P. W. Ellis, et al. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[130] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852, 2021.

[131] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725. IEEE, 2020.

[132] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Byol for audio: Self-supervised learning for general-purpose audio representation. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, Jul 2021.

[133] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Masked spectrogram modeling using masked autoencoders for learning general-purpose audio representation. In Joseph Turian, Björn W. Schuller, Dorien Herremans, Katrin Kirchoff, Paola Garcia Perera, and Philippe Esling, editors, *HEAR: Holistic Evaluation of Audio Representations (NeurIPS 2021 Competition)*, volume 166 of *Proceedings of Machine Learning Research*, pages 1–24. PMLR, 13–14 Dec 2022.

[134] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Masked Modeling Duo: Learning Representations by Encouraging Both Networks to Model the Input. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[135] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Masked Modeling Duo: Towards a Universal Audio Pre-training Framework. *IEEE/ACM Trans. Audio, Speech, Language Process.*, 32:2391–2406, 2024.

[136] Yuan Gong, Andrew Rouditchenko, Alexander H. Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James R. Glass. Contrastive audio-visual masked autoencoder. In *The Eleventh International Conference on Learning Representations*, 2023.

[137] Google. Webrtc. `https://www.webrtc.org/`. Accessed: 2026-01-23.

[138] Stefania Cecchi, Alessandro Terenzi, et al. A preliminary study of sounds emitted by honey bees in a beehive. In *Audio Engineering Society Convention 144*. Audio Engineering Society, 2018.

[139] Mayank Sharma, Sandeep Joshi, Tamojit Chatterjee, and Raffay Hamid. A comprehensive empirical review of modern voice activity detection approaches for movies and tv shows. *Neurocomputing*, 2022.

[140] John Wiseman. Python interface to the webrtc voice activity detector. `https://www.github.com/wiseman/py-webrtcvad/`, 2016. Accessed: 2026-01-23.

[141] Koji Okabe, Takafumi Koshinaka, and Koichi Shinoda. Attentive Statistics Pooling for Deep Speaker Embedding. In *Proc. Interspeech 2018*, pages 2252–2256, 2018.

[142] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, page 93–104, New York, NY, USA, 2000. Association for Computing Machinery.

[143] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008.

[144] Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W Schuller, Christian J Steinmetz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, et al. Hear: Holistic evaluation of audio representations. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 125–145. PMLR, 2022.

[145] Eklavya Sarkar and Mathew Magimai.-Doss. Comparing self-supervised learning models pre-trained on human speech and animal vocalizations for bioacoustics processing. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.

[146] Jason Ansel, Edward Yang, Horace He, et al. PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, April 2024.

[147] F Gene, B Emsen, and A Dodologlu. Effects of rearing period and grafting method on the queen bee rearing. *Journal of Applied Animal Research*, 27(1):45–48, 2005.

[148] Sara Ferrari, Mitchell Silva, Marcella Guarino, and Daniel Berckmans. Monitoring of swarming sounds in bee hives for early detection of the swarming period. *Computers and electronics in agriculture*, 64(1):72–77, 2008.

[149] Antonio Robles-Guerrero, Tonatiuh Saucedo-Anaya, Efrén González-Ramírez, and José Ismael De la Rosa-Vargas. Analysis of a multiclass classification problem by lasso logistic regression and singular value decomposition to identify sound patterns in queenless bee colonies. *Computers and Electronics in Agriculture*, 159:69–74, 2019.

[150] Stenford Ruvinga, Gordon J.A. Hunter, Olga Duran, and Jean-Christophe Nebel. Use of lstm networks to identify "queenlessness" in honeybee hives from audio signals. In *2021 17th International Conference on Intelligent Environments (IE)*, pages 1–4, 2021.

[151] Stefania Cecchi, Alessandro Terenzi, Simone Orcioni, et al. A preliminary study of sounds emitted by honey bees in a beehive. In *Audio Engineering Society Convention 144*. Audio Engineering Society, 2018.

[152] Inês Nolasco, Alessandro Terenzi, Stefania Cecchi, Simone Orcioni, Helen L Bear, and Emmanouil Benetos. Audio-based identification of beehive states. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8256–8260. IEEE, 2019.

[153] Koji Okabe, Takafumi Koshinaka, and Koichi Shinoda. Attentive statistics pooling for deep speaker embedding. In *Proc. Interspeech 2018*, pages 2252–2256, 2018.

[154] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14(8):2, 2012.

[155] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. Specaugment: A simple data augmentation method for automatic speech recognition. *Interspeech 2019*, Sep 2019.

[156] Mahsa Abdollahi, Yi Zhu, Heitor R. Guimarães, Nico Coallier, Ségolène Maucourt, Pierre Giovenazzo, and Tiago H. Falk. Urban: Urban beehive acoustics and phenotyping dataset. *arXiv preprint arXiv:submit/5647233*, 2024.

[157] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[158] Vladimir Kulyukin. Audio, image, video, and weather datasets for continuous electronic beehive monitoring. *Applied Sciences*, 11(10), 2021.

[159] Aditya Bhouraskar. Automation of feature selection and generation of optimal feature subsets for beehive audio sample classification. Master's thesis, Utah State University, 2020.

[160] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[161] Andrej Žgank. Acoustic monitoring and classification of bee swarm activity using mfcc feature extraction and hmm acoustic modeling. In *2018 ELEKTRO*, pages 1–4, 2018.

[162] Aleksandra Zlatkova, Zivko Kokolanski, and Dimitar Tashkovski. Honeybees swarming detection approach by sound signal processing. In *2020 XXIX International Scientific Conference Electronics (ET)*, pages 1–3, 2020.

[163] Harriet Hall, Martin Bencsik, and Michael Newton. Automated, non-invasive varroa mite detection by vibrational measurements of gait combined with machine learning. *Scientific Reports*, 13(1):10202, 2023.

[164] Anton Gradišek, Gašper Slapničar, Jure Šorn, Mitja Luštrek, Matjaž Gams, and Janez Grad. Predicting species identity of bumblebees through analysis of flight buzzing sounds. *Bioacoustics*, 26(1):63–76, 2017.

[165] Michael-Thomas Ramsey, Martin Bencsik, Michael Ian Newton, Maritza Reyes, Maryline Pioz, Didier Crauser, Noa Simon Delso, and Yves Le Conte. The prediction of swarming in honeybee colonies using vibrational spectra. *Scientific reports*, 10(1):9798, 2020.

[166] Axel Michelsen, Wolfgang H Kirchner, Bent Bach Andersen, and Martin Lindauer. The tooting and quacking vibration signals of honeybee queens: a quantitative analysis. *Journal of Comparative Physiology A*, 158:605–611, 1986.

[167] Heitor R Guimarães, Abhishek Tiwari, Mahsa Abdollahi, Anderson R Avila, and Tiago H Falk. Biome: A resource-efficient bioacoustic foundational model for iot applications. *arXiv preprint arXiv:2602.09970*, 2026.

[168] Yiyuan Yang, Kaichen Zhou, Niki Trigoni, and Andrew Markham. Ssl-net: A synergistic spectral and learning-based network for efficient bird sound classification. In *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 926–930, 2024.

[169] Alessandro Licciardi and Davide Carbone. Whalenet: A novel deep learning architecture for marine mammals vocalizations on watkins marine mammal sound database. *IEEE Access*, 12:154182–154194, 2024.

[170] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[171] Masato Hagiwara, Benjamin Hoffman, Jen-Yu Liu, Maddie Cusimano, Felix Effenberger, and Katie Zacarian. Beans: The benchmark of animal sounds. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[172] Abhishek Tiwari, Raymundo Cassani, Shruti Kshirsagar, Diana P Tobon, Yi Zhu, and Tiago H Falk. Modulation spectral signal representation for quality measurement and enhancement of wearable device data: A technical note. *Sensors*, 22(12):4579, 2022.

[173] Tiago H Falk and Wai-Yip Chan. Modulation spectral features for robust far-field speaker identification. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(1):90–100, 2009.

[174] Isao Tokuda, Tobias Riede, Jürgen Neubauer, Michael J Owren, and Hanspeter Herzel. Nonlinear analysis of irregular animal vocalizations. *The Journal of the Acoustical Society of America*, 111(6):2908–2919, 2002.

[175] Rianna Burnham. Animal calling behaviours and what this can tell us about the effects of changing soundscapes. In *Acoustics*, volume 5, pages 631–652. MDPI, 2023.

[176] Mahsa Abdollahi, Yi Zhu, Heitor R. Guimarães, Nico Coallier, Ségolène Maucourt, Pierre Giovenazzo, and Tiago H. Falk. Audio modulation spectral features for improved honeybee colony population prediction. *IEEE Sensors Journal*, pages 1–1, 2025.

[177] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024.

[178] Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[179] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[180] Adam Gazzaley and Anna C Nobre. Top-down modulation: bridging selective attention and working memory. *Trends in cognitive sciences*, 16(2):129–135, 2012.

[181] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780, 2017.

[182] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. Fsd50k: An open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852, 2022.

[183] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725, 2020.

[184] Heinrich Dinkel, Zhiyong Yan, Yongqing Wang, Junbo Zhang, Yujun Wang, and Bin Wang. Scaling up masked audio encoder learning for general audio classification. In *Interspeech 2024*, 2024.

[185] Xuyao Deng, Tianjiao Wan, Kele Xu, Tian Gao, Peng Qiao, Dawei Feng, and Yong Dou. Scaling bioacoustic signal pre-training with million samples via mask-modeling. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.

[186] Heitor R. Guimarães, Arthur Pimentel, Anderson R. Avila, Mehdi Rezagholizadeh, Boxing Chen, and Tiago H. Falk. Robustdistiller: Compressing universal speech representations for enhanced environment robustness. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[187] Jianhua Zhang, Yi Gao, Ruyu Liu, Xu Cheng, Houxiang Zhang, and Shengyong Chen. Can students beyond the teacher? distilling knowledge from teacher's bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22434–22442, 2025.

[188] Titouan Parcollet, Rogier van Dalen, Shucong Zhang, and Sourav Bhattacharya. Linear time complexity conformers with summarymixing for streaming speech recognition. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.

[189] Théodor Lemerle, Nicolas Obin, and Axel Roebel. Small-e: Small language model with linear attention for efficient speech synthesis. In *Proc. Interspeech 2024*, pages 3420–3424, 2024.

[190] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2023.

[191] Guan-Ting Lin, Shang-Wen Li, and Hung-yi Lee. Listen, adapt, better wer: Source-free single-utterance test-time adaptation for automatic speech recognition. In *Proc. Interspeech 2022*, pages 2198–2202, 2022.

[192] Tobias Raichle, Niels Edinger, and Bin Yang. Test-time adaptation for speech enhancement via domain invariant embedding transformation. *IEEE Open Journal of Signal Processing*, pages 1–10, 2026.

[193] Camilla Casula and Sara Tonelli. On the impact of hate speech synthetic data on model fairness. In *Proceedings of the Eleventh Italian Conference on Computational Linguistics (CLiC-it 2025)*, pages 194–205, 2025.

[194] Naorem Karline Singh, Wangkheimayum Madal, Chingakham Neeta Devi, Hoomexsun Pangsatabam, and Yambem Jina Chanu. Leveraging synthetic data for improved manipuri-english code-switched asr. *IEEE Access*, 13:25723–25740, 2025.

[195] Anthony Gibbons, Emma King, Ian Donohue, and Andrew Parnell. Generative ai-based data augmentation for improved bioacoustic classification in noisy environments. *arXiv preprint arXiv:2412.01530*, 2024.

[196] Benjamin Hoffman, David Robinson, Marius Miron, Vittorio Baglione, Daniela Canestrari, Damian Elias, Eva Trapote, Felix Effenberger, Maddie Cusimano, Masato Hagiwara, et al. Synthetic data enables context-aware bioacoustic sound event detection. *arXiv preprint arXiv:2503.00296*, 2025.

[197] Kumudu Samarappuli, Iman Ardekani, Mahsa Mohaghegh, and Abdolhossein Sarrafzadeh. Markov chain wave generative adversarial network for bee bioacoustic signal synthesis. *Sensors*, 26(2):371, 2026.