

NOVEL AI-ASSISTED RESPIRATORY SOUND ANALYSIS FOR CLINICAL AND HOME-BASED HEALTH MONITORING

Thanh Dat Ngo

a thesis submitted for the degree of
Doctor of Philosophy in Computer Science

School of Computer Science and Electronic Engineering
University of Essex

April 2026

Acknowledgements

I would like to extend my sincere gratitude to my supervisors, Dr. Delaram Jarchi and Professor Klaus McDonald-Maier. Their guidance and advice were instrumental to my research and a constant source of support throughout my PhD. I am especially thankful to Dr. Delaram Jarchi, who is always believing in my ability to overcome challenges and providing invaluable insights that enriched my research. Her mentorship was pivotal to my academic and personal development. I am deeply grateful to my colleague Dr. Lam Pham for his great support and thoughtful guidance. I also sincerely thank all my PhD friends Stefan and Ibby for the wonderful workshops we shared, which brought not only valuable insights but also laughter and a great sense of joy. Finally, I want to thank my colleagues and friends in the Intelligent Systems (EIS) Research Laboratory for making it a welcoming and supportive place to work.

Abstract

This thesis investigates the classification of respiratory anomalies using advanced audio feature extraction and deep learning techniques within the framework of Classification of Respiratory Audio Anomalies (CRAA). The study examines how different time–frequency representations and deep learning methods can enhance the detection and classification of abnormal respiratory sounds. By utilizing multiple spectrogram and scalogram representations, along with deep learning approaches such as joint learning models, ensemble systems, attention-based architectures, Inception–Residual networks, and an embedding-based alignment framework for cross-domain CRAA, the research demonstrates that respiratory anomalies can be detected with high accuracy and robustness, even in challenging clinical and home-based environments. Overall, the findings highlight the potential of audio-based deep learning systems as non-invasive, scalable, and cost-effective tools for the early detection and monitoring of respiratory diseases. The results support the development of practical respiratory screening solutions that can be integrated into telehealth platforms and mobile health applications, particularly in resource-limited settings. Ultimately, this work aims to enhance diagnostic support and help reduce the global burden of respiratory diseases, especially in regions with limited access to skilled healthcare professionals.

Contents

Acknowledgements	ii
Abstract	iii
Contents	iv
List of Tables	ix
List of Figures	xii
1 Introduction	1
1.1 Classification of Respiratory Audio Anomalies (CRAA)	1
1.2 Key Challenges in CRAA	3
1.2.1 Audio Feature Extraction	5
1.2.2 Deep Learning Classification	6
1.2.3 Cross-domain Classification from Clinical Environments . .	7
1.2.4 Summary of Key Challenges in CRAA	8
1.3 Research Questions (RQs)	9
1.4 Research Objectives (ROs)	9
1.5 Thesis Contributions	10
1.6 List of Publications	12
1.7 Thesis Structure	13

2	Literature Review	16
2.1	Audio Feature Extraction in CRAA	16
2.1.1	Hand-crafted Feature	16
2.1.2	Frequency and Time Representation	18
2.2	Artificial intelligence (AI) models in CRAA	20
2.2.1	Traditional Machine Learning	20
2.2.2	Deep Learning	20
2.3	Evaluation Metrics applied in CRAA	22
2.3.1	Sensitivity	23
2.3.2	Specificity	23
2.3.3	Average Score	23
2.3.4	Harmonic Score	23
2.3.5	Score	23
2.3.6	Accuracy	23
2.3.7	Summary of Evaluation Metrics in CRAA	24
2.4	Benchmark Datasets in CRAA	25
2.4.1	ICBHI 2017	25
2.4.2	HF-Lung-v1	26
2.4.3	SPRSound 2023	27
2.4.4	Coswara Dataset	28
2.4.5	Dataset Conclusions	28
2.5	Summary	30
3	Audio Feature Extraction	32
3.1	Short Time Fourier Transform	32
3.1.1	Log-Mel	33
3.1.2	MFCC	33
3.1.3	CQT	34
3.1.4	Gammatone	35

3.2	Foundational approach on Multi-spectrogram Features and Joint Learning Models	36
3.2.1	Acoustic Datasets and Setting	38
3.2.2	High-Level Architecture and Baseline C-DNN Network Proposed	39
3.2.3	An Analysis of Spectrograms	40
3.2.4	Late Fusion Strategy to Evaluate Spectrogram Combinations	43
3.2.5	Joint Learning Deep Neural Network Architecture Proposed	47
3.3	Continuous Wavelet Transform	52
3.3.1	Analytic Morlet (Gabor) Wavelet (Amor)	53
3.3.2	Generalized Morse Wavelet (Morse)	54
3.3.3	Bump-shaped frequency response wavelet (Bump)	55
3.4	Summary on Feature Extraction	56
4	Proposed Deep Learning Models	57
4.1	Deep Learning Framework Applied For Predicting Anomaly of Respiratory Sounds	57
4.1.1	ICBHI Dataset and Task Definition	57
4.1.2	Front-End Feature Extraction	59
4.1.3	Proposed Encoder-Decoder Network Architecture	61
4.1.4	Experimental Setting	62
4.1.5	Results and Performance	65
4.2	An Ensemble of Deep Learning Frameworks Applied for Predicting Respiratory Anomalies	66
4.2.1	The Low-footprint Inception Based Network architectures	68
4.2.2	The Benchmark and Complex Neural Network Architectures	69
4.2.3	The Transfer Learning Based Network Architectures	70
4.2.4	Early, Middle, and Late Fusion of Inception Based and Transfer Learning Based Frameworks	71

4.2.5	Results and Performance	73
4.3	An Inception-Residual-Based Architecture with Multi-Objective Loss for Detecting Respiratory Anomalies	74
4.3.1	SPR Dataset and Task Defined	74
4.3.2	Front-End Feature Extraction	76
4.3.3	Inception-residual-based Backbone Models	78
4.3.4	Multi-Objective Loss	82
4.3.5	Results and Performance	82
4.4	A Deep Learning Architecture with Spatio-Temporal Focusing for Detecting Respiratory Anomalies	85
4.4.1	IEEE BioCAS 2023 grand challenge and Tasks Defined . .	85
4.4.2	Data Augmentation	87
4.4.3	Spatio-Temporal Focusing layers	88
4.4.4	Results on Event Level	91
4.4.5	Results on Recording Level	91
5	Cross Domain Adaptation	97
5.1	Audio-Based Deep Learning Frameworks for Detecting COVID-19	97
5.1.1	Coswara Dataset and Task Defined	98
5.1.2	The Proposed Framework	98
5.1.3	The Low-level Spectrogram Feature Extraction	99
5.1.4	The High-level Embedding Feature Extraction	99
5.1.5	Back-end Classification	102
5.1.6	Results and Performance	103
5.2	An Efficient Embedding Alignment Framework for Detecting Res- piratory Anomalies Using Deep Learning	109
5.2.1	Front-End Feature Extraction	110
5.2.2	Deep Learning Backbone Architecture	110
5.2.3	Proposed Feature Alignment Framework	112

5.2.4	Experimental Setup and Results	116
6	Conclusion and Future Work	127
6.1	Overview	127
6.2	Summary of Contributions and Significance	130
6.3	Impact on Respiratory Disease Detection	131
6.4	Future Directions	133
6.5	Closing Remarks	134
A	Mathematical definitions	136
A.1	Computation of Network Layers	136
A.1.1	Rectify Linear Unit (ReLU) Layer	136
A.1.2	Dropout Layer	137
A.1.3	Batch Normalization Layer	137
A.1.4	Fully Connected Layer	138
A.1.5	Softmax Layer	138
A.1.6	Convolutional Layer	138
A.1.7	Multi-head Attention Layer	139
	Bibliography	140

List of Tables

1	Characteristics of Abnormal Lung Sounds: Wheeze and Crackle. .	2
2	Details of information on datasets used in this work.	31
3	Network layers used in C-DNN architecture.	40
4	Spectrogram combinations proposed.	42
5	Compare individual spectrograms and their combinations to DCASE 2018 Task 1B baseline with best results (%) in bold	46
6	C-RNN network architecture.	47
7	DNN-02 network architecture.	49
8	Performance comparison (%) on DCASE 2018 Task 1B dataset with the highest scores highlighted in bold	51
9	Comparison of the state-of-the-art systems with best performance highlighted in bold	51
10	Confusion matrix of anomaly cycle classification.	58
11	Feature Extraction Parameter Setting.	59
12	C-DNN network architecture.	61
13	Encoder-decoder network architecture.	63
14	MLP-based network architecture.	63
15	ICBHI dataset splitting.	64
16	Performance of C-DNN, Autoencoder, and their fusions (highest scores in bold).	66

17	Compare our systems (the lower part) against state-of-the-art systems with ICBHI challenge splitting (highest scores in bold). . . .	66
18	The general inception-based network architectures.	69
19	Settings for inception-based network architectures.	70
20	Performance comparison of proposed deep learning frameworks with model size.	93
21	The MLP architecture used for training embedding features. . . .	93
22	Performance comparison of fusion strategies of inception based and transfer learning based frameworks.	94
23	Comparison against the state-of-the-art systems.	94
24	Performance comparison on WA, GA, and WM branches on the official test set in SPRSound dataset 2022.	94
25	Performance comparison on System I, II, and III on the official test set in SPRSound dataset 2022.	95
26	Performance of our proposed system compared to others on the official test set in SPRSound dataset 2022 (Score(%)).	95
27	Performance comparison among spectrograms on the Validation set in Task 1-1 and Task 1-2 (event level).	95
28	Performance comparison among spectrograms on Validation set in Task 2-1 and Task 2-2 (entire recording level).	96
29	Compare our proposed systems to the others submitted IEEE Bio-CAS 2023 challenge (Score(%) on the blind Test set).	96
30	Configuration of Lenet based architecture.	101
31	Comparison of low-level spectrograms (AUC/SE).	102
32	Comparison of pre-trained deep learning network architectures directly trained on DiCOVA dataset (AUC/SE).	104
33	Comparison of pre-trained models trained on AudioSet dataset (AUC/SE).	106

34	Performance comparison with the state-of-the-art systems on Test set (Cough and Speech).	108
35	Performance comparison with the state-of-the-art systems on Test set (Breathing and All).	108
36	Performance comparison across different back-end classification models on Test set (AUC/SE).	109
37	Training sample distribution across datasets	117
38	Compare our deep learning backbone to the others submitted to the IEEE BioCAS 2023 challenge.	119
39	The performance of source and target domains is evaluated using feature alignment, direct transfer, and training from scratch. . . .	122
40	Feature alignment performance by recording devices on target datasets (ICBHI and HF_Lung_v1).	126

List of Figures

1	The high-level architecture of our proposed system for detecting anomalies in respiratory audio inputs.	3
2	High-level architecture of our Acoustic Scene Classification system.	37
3	Block-level architecture of C-DNN network.	39
4	Constructed spectrogram tree basing on difference of central frequencies and auditory models applied.	41
5	Category-wise performance comparison among spectrograms on device A - DCASE 2018 Task 1B.	42
6	Category-wise performance comparison among spectrograms on device B&C - DCASE 2018 Task 1B.	45
7	Joint learning network architecture.	48
8	The Frequency and Time resolution in three mother wavelet functions: Morse (Top), Amor (Middle), and Bump (Bottom)).	53
9	Audio signals and Scalograms of Normal, Wheeze and Crackles.	54
10	Deep learning based framework proposed.	60
11	Block-level architecture of Autoencoder network.	61
12	The high-level architecture for three deep learning frameworks proposed.	68
13	The single inception layer architecture.	70
14	The high-level architecture of our proposed system for detecting anomalies in respiratory audio recording.	76

15	The detailed architecture of the Backbone Model used in the proposed system.	77
16	The efficiency of applying contrastive loss in the training process on Task 1-2.	82
17	The high-level architecture of our proposed system for detecting anomalies in respiratory audio inputs.	86
18	The proposed architecture for the Back-end classification.	88
19	The high-level architecture of the proposed frameworks.	99
20	Influence of over-sampling on positive samples and dimensionality reduction within Track-4 DiCOVA Challenge.	105
21	The detailed architecture of deep learning backbone architecture.	112
22	The Feature Alignment Framework with two main Source and Target streams.	114
23	A visual summary of key findings on ICBHI and HF_lung_v1 dataset from Table 39.	120
24	The t-SNE visualization: Embedding features extracted from Combiner block in Target Domain Stream.	121

Chapter 1

Introduction

1.1 Classification of Respiratory Audio Anomalies (CRAA)

Chronic respiratory diseases, including tuberculosis, asthma, chronic obstructive pulmonary disease (COPD), and lower respiratory tract infections (LRTIs), contribute to a significant global health burden, resulting in over 6.2 million deaths annually. In the UK, chronic respiratory illnesses cause the hospitalization of over 700,000 people every year [1]. These numbers are expected to rise due to the lack of accessible, tailored tools for early diagnosis and intervention, particularly in under-resourced regions. Recognizing this critical gap, this thesis proposes an AI-assisted diagnostic component that analyzes respiratory sounds to identify abnormal patterns. By integrating conventional machine learning and deep learning techniques, CRAA enables the segmentation and classification of various lung sound events, supporting both clinicians and end-users in detecting potential health issues at an early stage.

In detail, respiratory sounds are generated by the movement of air through the respiratory tract, and can be categorized as normal or abnormal. Normal

sounds, such as tracheal, bronchial, bronchovesicular, and vesicular sounds, are produced by healthy lungs. Abnormal sounds, on the other hand, are associated with lung or bronchial diseases. These abnormal sounds can be characterized by their underlying cause, location, and acoustic properties, such as their duration, pitch, and timing [2].

Table 1: Characteristics of Abnormal Lung Sounds: Wheeze and Crackle.

Feature	Wheeze	Crackle
Mechanism of sound production	Airway narrowing or obstruction.	The sudden opening of collapsed small airways and alveoli.
Acoustic Properties		
Frequency Range	100 Hz – 2000 Hz. [2, 3]	350 Hz – 650 Hz. [2, 3]
Duration	Continuous, typically ≥ 80 ms. [2, 3, 4]	Discontinuous, typically 5 ms – 15 ms. [2, 3, 4]
Acoustic Characteristics		
Waveform	Continuous, musical, sinusoidal.	Discontinuous, sharp, rapidly dampened wave deflection.
Related Respiratory Disease		
Common Conditions	Asthma, Chronic Obstructive Pulmonary Disease (COPD), Endobronchial tumor, Foreign body in bronchus.	Chronic bronchitis, Pneumonia, Pulmonary Edema, Idiopathic pulmonary fibrosis, Congestive heart failure.

The characteristics of abnormal sounds are provided in Table 1. In details, crackles, for instance, are short, explosive sounds heard during inhalation and sometimes exhalation [2, 3, 4]. They can be further classified into fine and coarse crackles, each associated with specific lung conditions such as chronic bronchitis and idiopathic pulmonary fibrosis. Meanwhile, wheezes, high-pitched sounds with a sinusoidal oscillation, often occur in narrowed or obstructed airways, as seen in

diseases such as asthma and COPD [3].

Understanding these sound patterns is crucial for automated analysis in CRAA. Accordingly, most proposed systems for detecting anomalies in respiratory sounds as shown in Figure 1 consists two main stages: low-level spectrogram feature extraction and back-end classification.

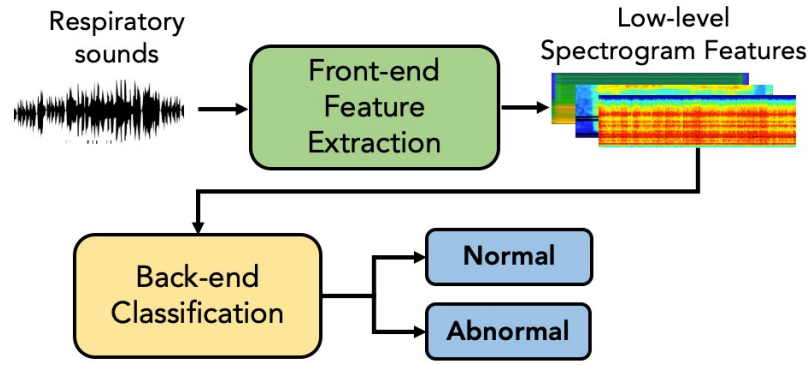


Figure 1: The high-level architecture of our proposed system for detecting anomalies in respiratory audio inputs.

This thesis analyzes the related work of two key stages in a deep learning-based system for detecting anomalies from respiratory sounds: The front-end feature extraction using spectrograms and the back-end deep neural network architecture. We then propose techniques to improve the performance of our systems.

1.2 Key Challenges in CRAA

Many studies have evaluated human ability to auscultate and identify respiratory sounds. The major drawback of respiratory auscultation is its inherent subjectivity [5, 6]. The work in [6, 7] found that pulmonologists performed better than other doctors and medical students in auscultation. Meanwhile, Melbye et al [8] and Mangione et al [9] reached the same conclusion that internal medicine and family practice trainees performed no better than medical students, while

pulmonary physicians scored the highest. In contrast, Mehmood et al. [10] reported that attending physicians in intensive care units outperformed nurses and pulmonary physicians in auscultatory accuracy. Overall, these studies highlight the discrepancies in auscultation ability among all participants, which may lead to inaccurate diagnoses or treatments. Another limitation of auscultation is the inability to provide remote assessment. When using traditional stethoscopes, doctors must physically place it on the patient's chest wall. However, many patients with chronic conditions or limited mobility often reside in nursing homes or at home without regular access to healthcare providers. As a result, it remains challenging for doctors to perform auscultation on each individual. Therefore, there is a growing demand for telemedicine among patients in remote areas.

To reduce subjectivity and to remotely monitor respiratory conditions for patients, a user-friendly technology such as a digital stethoscope with data transmission capability to record and share auscultated sounds would be highly beneficial. Especially for remote care and the reduction in the dependence of auscultation on physical presence [11]. However, some recorded respiratory sounds remain ambiguous which cause even trained clinicians and experts not to discern well. This ambiguity can arise from two main factors. First, the differences in built-in amplification systems, cross-channel mismatch from different recording devices with different sampling rates, and the absence of standardized methods for minimizing environmental and physiological noise that significantly alter the quality of recordings [11, 12]. Second, the majority of participants in most databases are adult, which inevitably cause biased and subjective differences in paediatric respiratory sound analysis. These problems have been highlighted in [13, 14] and now is illustrating the need for comprehensive digital auscultation on both paediatric and adult respiratory sounds.

To the best of our knowledge, this is the first study in the literature, which

applies deep learning architecture and a cross-domain adaptive method for detecting anomalies in respiratory sound auscultation among current published datasets. It can be a valuable tool in addressing the variability and ensuring that models trained on one dataset can generalize effectively to other datasets. By aligning features across different age groups of patients, recording environments and conditions, our proposed approach can enhance the robustness of respiratory sound analysis, leading to more accurate and reliable diagnostic tools. In this study, we evaluate respiratory auscultation across currently available datasets. Our proposed deep learning embedding alignment approach requires three main factors: The front-end feature extraction using scalograms, the backbone deep neural network architecture, and the embedding feature alignment method.

1.2.1 Audio Feature Extraction

In recent years, automated analysis of lung sounds has gained significant attention for its potential to assist in the diagnosis of respiratory diseases. Existing approaches can broadly be categorized into two main groups based on the type of input features used.

The first group relies on hand-crafted features derived from the raw audio signals. Techniques such as statistical measures [15], entropy-based features [16], and Mel Frequency Cepstral Coefficients (MFCCs) [17, 18, 19, 20] are commonly employed to convert lung sounds into discriminative feature vectors. These features are then utilized by traditional machine learning models such as Hidden Markov Models (HMMs) [19], Support Vector Machines (SVMs) [21], and Decision Trees [18] for tasks like classification and anomaly detection.

The second group, on the other hand, adopts a more data-driven approach

by transforming audio signals into two-dimensional spectrograms, including S-transform [22], Short-time Fourier Transform (STFT) spectrogram [23], and Log-Mel spectrogram [24, 25, 26], which encode both spectral and temporal characteristics. Other spectrograms include Gammatonegram [27, 28], and Constant-Q Transform (CQT) spectrogram [28, 29, 30]. These representations are then fed into deep learning architectures such as Convolutional Neural Networks (CNNs) [31, 32] or Recurrent Neural Networks (RNNs) [33, 34] for end-to-end classification.

1.2.2 Deep Learning Classification

As regards the deep neural network architecture, a wide range of models have been employed to improve the accuracy and robustness of lung sound classification systems. Among them, convolutional neural networks (CNNs) remain the most commonly adopted due to their strong performance in capturing spatial patterns from spectrogram representations of audio signals [26, 32, 35, 36]. These networks are effective in identifying localized features in time-frequency domains, making them suitable for respiratory sound analysis. Beyond standard CNNs, more advanced and hybrid models have been proposed to address the complex nature of lung sound signals. For instance, Acharya et al. [30] introduced a hybrid CNN-RNN deep learning model that integrates the spatial feature extraction capabilities of CNNs with the strength of recurrent neural networks (RNNs) in modeling temporal sequences. This architecture enables the system to capture both frequency-related patterns and temporal dynamics, leading to improved classification performance. On the other hand, Truc et al. [37] leveraged pre-trained ResNet models, a form of deep residual networks, as the foundational architecture. Their approach incorporates various transfer learning techniques such as fine-tuning, co-tuning, and stochastic normalization to enhance generalization and optimize

performance across diverse datasets. This strategy proved to be highly effective, outperforming traditional methods for the classification of adventitious lung sounds and the diagnosis of respiratory diseases. Generally, these studies highlight the growing trend towards leveraging deeper and more sophisticated neural architectures, often combined with domain adaptation techniques, to achieve superior results in automatic lung sound classification.

1.2.3 Cross-domain Classification from Clinical Environments

Domain adaptation in deep learning has become increasingly crucial in medical signal analysis, particularly where the training and deployment processes are often downgraded in performance due to different distributions. In the context of lung sound classification, such domain discrepancies can arise from variations in patient age groups, recording environments, or sensor devices. Existing deep domain adaptation methods can be broadly categorized into three groups [38, 39, 40]. The first and most common approach—fine-tuning—adjusts pre-trained models on target data, but often at the cost of forgetting source domain knowledge. The second group, weighted local domain adaptation methods [41, 42], attempts to mitigate this by re-weighting samples to approximate the target distribution. However, it remains sensitive to the choice of weighting strategy. The third and most promising category, embedding-based alignment [38, 40], transforms data from both source and target domains into a shared latent space where distributional alignment can be achieved. Inspired by this approach, we propose a novel embedding-based domain adaptation method that aligns class-wise instance distributions between domains. By mapping input data into a structured representation space through a fully connected neural network, our method assumes and enforces local geometric consistency among same-class instances across domains. This alignment strategy enables more robust cross-domain generalization

in the classification of chest wall lung sounds. For example, this strategy enhances cross-domain generalization, enabling more accurate and reliable lung sound classification in diverse clinical settings.

1.2.4 Summary of Key Challenges in CRAA

In this thesis, we address the following key challenges:

- **Robustness across diverse clinical environments:** Respiratory sound recordings vary substantially due to differences in patient populations (e.g., adult versus pediatric), recording settings, and acquisition devices. These variations create domain shifts that reduce the performance of specific conventional classification models when applied to new datasets. As a result, models trained on one clinical dataset often fail to generalize reliably to others. Addressing this issue requires domain adaptation strategies that can learn invariant and transferable representations across different clinical conditions.

- **Effective and consistent representation learning for lung sound analysis:** Respiratory recordings are frequently contaminated by background noise, ambient interference, and motion artifacts, which can obscure clinically relevant acoustic patterns in crackles and wheezes. Designing suitable feature representations also involves a trade-off between temporal and spectral resolution: short windows may miss broader spectral structure, while longer windows may fail to capture transient events. Furthermore, high-dimensional audio representations increase computational cost and the risk of overfitting, particularly when annotated data are limited. Developing compact, noise-robust, and discriminative representations is therefore essential for reliable respiratory sound classification.

These challenges collectively motivate the development of the domain-adaptive and representation learning methods proposed in this thesis, which aim to improve robustness, generalization, and clinical applicability of CRAA systems.

1.3 Research Questions (RQs)

This section outlines the key research questions that the thesis aims to explore:

- RQ1: While various spectrogram types have been proposed—typically used as input features for deep learning models—they often outperform traditional hand-crafted features. However, no comprehensive study has yet addressed how to select the most suitable spectrogram type, or whether a particular kind is consistently effective for lung sound analysis.

- RQ2: Despite the growing success of deep learning techniques in biomedical signal processing, identifying a robust model architecture that performs well for lung disease detection remains an open problem. Can a unified deep learning classifier be developed that maintains strong performance under real-world noise conditions such as speech, coughing, motion artifacts, and other physiological sounds?

- RQ3: Respiratory sound datasets often vary significantly due to differences in patient demographics (e.g., adult versus pediatric), recording environments, and the types of devices used. These domain shifts lead to performance degradation when conventional models are applied to datasets beyond the training domain. How can domain adaptation be effectively applied to enable generalizable performance across diverse clinical conditions and recording scenarios?

- RQ4: Given the high computational cost of deep learning frameworks, is there an efficient strategy to reduce the model’s memory footprint without substantially compromising accuracy in lung sound classification tasks?

1.4 Research Objectives (ROs)

To address the above research questions, this thesis pursues the following objectives:

- RO1: To systematically analyze and compare multiple spectrogram representations and investigate the effectiveness of multi-spectrogram feature fusion. (Addresses RQ1)
- RO2: To design and evaluate robust deep learning architectures tailored for respiratory sound classification under real-world noise conditions. (Addresses RQ2)
- RO3: To improve feature discriminability, training stability, and interpretability through multi-objective loss functions and spatio-temporal attention mechanisms. (Further, it supports RQ1–RQ3 and strengthens the overall framework robustness)
- RO4: To design computationally efficient model architectures and optimization strategies that reduce memory usage while preserving classification performance. (Addresses RQ4)
- RO5: To develop and integrate domain adaptation and embedding alignment strategies to enhance cross-domain generalization. (Addresses RQ3)

1.5 Thesis Contributions

The primary contributions of this thesis are outlined below:

1. Design and evaluation of multiple deep learning architectures for respiratory sound analysis. This thesis introduces and evaluates a suite of deep learning models tailored to classify respiratory sounds. These models include joint-learning architectures, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks with specific challenges in respiratory anomaly detection from audio signals. (Outcome from RO1, RO2 and RO5)

2. Present the utilization of multi-spectrogram features for comprehensive audio representation. The thesis advocates a multi-spectrogram approach by combining representations such as Log-mel, STFT, MFCC, Gammatone and CQT spectrograms. This strategy demonstrate the use of multi-spectrogram can enhances the diversity and richness information of time-frequency features. (Outcome from RO1)
3. Introduce the development of a multi-objective loss function. A novel balanced loss function was introduced to simultaneously optimize multiple learning objectives, including classification accuracy, combined feature embeddings, and contrastive learning separability. This optimization led to more stable training and better generalization. (Outcome from RO3)
4. Implement an ensemble learning techniques to enhance predictive reliability. In details, an ensemble of diverse deep learning models was constructed to aggregate their probability outputs. This approach reduces predictive variance and improves overall robustness for respiratory sound classification. (Outcome from RO2)
5. Propose a spatio-temporal attention mechanism for targeted feature focus. A spatio-temporal attention module was integrated into the model pipeline, enabling dynamic focus on the most informative regions within spectrograms. This not only improved classification accuracy but also enhanced interpretability. (Outcome from RO1 and RO2)
6. Implement a wide range of audio-based deep learning frameworks applied to the breathing, cough, and speech sounds for detecting COVID-19. Conducting extensive experiments and pinpoint the most effective approach related to the extraction of well-represented features for each type of breathing, cough, speech sound input. The framework is reliable and robust for detecting COVID-19 which outperforms the state-of-the-art systems and is

potential for a real-life application. (Outcome from RO1 and RO2)

7. Present an embedding alignment framework for discriminative representation learning to refine latent feature representations. This method encourages more discriminative embeddings, improving the separability across varied recording conditions by clinical environments and patient demographics. (Outcome from RO5)

1.6 List of Publications

This section shows published and preprint papers that are relevant to and contribute into this thesis.

Conference publications as first author:

1. **Dat Ngo**, Huy Phan, Minh Tran, Delaram Jarchi and Şefki Kolozali, “An Inception-Residual-Based Architecture with Multi-Objective Loss for Detecting Respiratory Anomalies”, in IEEE 25th International Workshop on Multimedia Signal Processing (MMSP), pp. 1301-1306. IEEE, 2023 [35]. (Referred to RO3)
2. **Dat Ngo**, Lam Pham, Huy Phan, Minh Tran, Delaram Jarchi, “A Deep Learning Architecture with Spatio-Temporal Focusing for Detecting Respiratory Anomalies”, in Proc. IEEE Biomedical Circuits and Systems Conference (BioCAS), pp. 1201-1205. IEEE, 2023 [29]. (Referred to RO3)
3. **Dat Ngo**, Lam Pham, Truong Hoang, Şefki Kolozali, and Delaram Jarchi, “Audio-based deep learning frameworks for detecting COVID-19”, in 2022 30th European Signal Processing Conference (EUSIPCO), pp. 1233-1237. IEEE, 2022 [25]. (Referred to RQ2)
4. **Dat Ngo**, Yordan Raykov, Mahdi Shaban, Delaram Jarchi, Arshia Gratiot, Chas Sheppard, “Advancements in early airway disease detection using

vowel phonation with smartphone”, in European Respiratory Journal Open Research (ERJ Open Research), vol. 10, 2024 [43]. (Referred to RO2 and RO4)

Journal publications as first author:

1. **Dat Ngo**, Lam Pham, Anh Nguyen, Tien Ly, Khoa Pham, Thanh Ngo, “Sound context classification basing on join learning model and multi-spectrogram features”, in International Journal of Computing, vol. 21, 2022 [44]. (Referred to RO1)
2. **Dat Ngo**, Lam Pham, Sefki Kolozali, Klaus D. McDonald-Maier, Delaram Jarchi, “An Efficient Embedding Alignment Framework for Detecting Respiratory Anomalies Using Deep Learning”, under review, in IEEE Transactions on Biomedical Engineering (TBME), 2025. (Referred to RO5)

Other related publications:

1. L. Pham, **Dat Ngo**, Khoa Tran, Truong Hoang, Alex Schindler and Ian McLoughlin, “An Ensemble of Deep Learning Frameworks for Predicting Respiratory Anomalies”, in 2022 44th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 4595-4598. IEEE, 2022 [45].
2. **Dat Ngo**, Lam Pham, Anh Nguyen, Ben Phan, Khoa Tran, and Truong Nguyen, “Deep learning framework applied for predicting anomaly of respiratory sounds”, in 2021 International Symposium on Electrical and Electronics Engineering (ISEE), pp. 42-47. IEEE, 2021 [32].

1.7 Thesis Structure

These remaining chapters of this thesis are organized as follows.

Chapter 2 presents a comprehensive review of recent research on CRAA, including the latest systems and newly released datasets. Through detailed analysis, several open challenges in the CRAA task are identified and discussed.

Chapter 3 provides a comprehensive analysis on spectrograms by conducting experiments on five common types of spectrograms, comprising of STFT, Log-Mel, MFCC, CQT, and Gammatone. However, their reliance on fixed window sizes during Fast Fourier Transform (FFT) limits their ability to capture fine-grained spectral and temporal information, which is essential to represent the distinguishing normal from abnormal respiratory sounds. To address this limitation, we propose an alternative approach using multiple scalograms derived from Continuous Wavelet Transform (CWT), employing Amor, Bump, and Morse as the mother wavelet functions. This approach not only generates a better multi-resolution analysis thanks to its suitability in adjusting in temporal window length, but also shows great ability to capture the energy of the sound signals in wider frequency range, which is strongly relevant to the nature of respiratory sounds.

Chapter 4 proposes comprehensive works conducted on exploring the role of deep learning-based architectures on two-dimensional spectrograms based on feature extraction methods in Chapter 3. In details, we design and evaluate multiple deep learning architectures tailored to CRAA. These models include joint-learning architectures, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks with specific challenges in respiratory anomaly classification from audio signals.

Chapter 5 proposes domain adaptation techniques with previous deep learning architectures presented in Chapter 4 to address the significant variability in data quality of respiratory sound recordings. To the best of our knowledge, this is the first study in the literature, which applies a cross-domain adaptive method for detecting anomalies in respiratory sound auscultation using all currently published

datasets. Our proposed method can be a valuable tool in addressing the variability and ensuring that models trained on one dataset can generalize effectively to other datasets. By aligning features across different age groups of patients, recording environments and conditions, our proposed domain adaptation approach can enhance the robustness of respiratory sound analysis, leading to more accurate and reliable diagnostic tools. We perform the domain adaptation in respiratory auscultation across datasets published to date, our proposed deep learning embedding alignment approach requires three main factors: The front-end feature extraction using scalograms, the backbone deep neural network architecture, and the embedding feature alignment method.

Chapter 6 concludes this thesis by presenting a conclusion and outlining future work.

Appendix where computation of spectrograms and network layers are described in detail.

Chapter 2

Literature Review

2.1 Audio Feature Extraction in CRAA

Audio Feature Extraction (AFE) is one of two main steps in a system used for sound analysis where raw audio is transformed into low-level features that capture rich and condensed information. State-of-the-art audio feature extraction used for lung sounds are separated into two main groups including hand-crafted features and spectrogram(in form of Frequency and Time representation).

2.1.1 Hand-crafted Feature

As the first group making use of hand-crafted features, a variety of techniques such as statistical features [15, 19, 21], entropy-based features [16, 18], and Mel Frequency Cepstral Coefficients (MFCCs) [17, 18, 19, 20] are exploited to transform lung sounds into feature vectors. Next, these feature vectors are then processed by conventional machine learning models to classify anomalies. However, these hand-crafted features often rely heavily on expert knowledge, and may struggle to generalize across diverse recording conditions or patient populations, limiting

their robustness and scalability [18, 28]. Furthermore, while hand-crafted features such as MFCCs and other statistical features are employed to represent general frequency properties, respiratory sounds including crackles and wheezes have unique acoustic markers that need to be captured both the time and frequency domain. First, research from [17, 18, 28] has demonstrated that the reliance on traditional hand-crafted audio features, such as Mel-Frequency Cepstral Coefficients (MFCCs) and entropy-based features, presents insignificant performance when classifying subtle respiratory sound anomalies. Primarily, these features are based on predefined mathematical models and established domain-specific knowledge, often inherited from speech recognition applications. This fixed methodology makes it less effective to the diverse range of real-world noise invariably presented in clinical environments. This includes artifacts like patient movement or ambient room sounds, which is often non-stationary and obscures the subtle diagnostic signals. Because MFCCs and similar features are not dynamically optimized to filter or adapt to this noise, they frequently lead to the extraction of irrelevant or misleading acoustic information, compromising the the performance of back-end classifiers. Second, hand-crafted features are frequently struggle to adequately capture the complex, non-linear, and non-stationary nature of adventitious lung sounds, such as crackles and wheezes. These sounds are dynamic, and the calculation process for traditional features tends to average out or lose crucial temporal dependency information. While the precise in timing and frequency information of the sound bursts are vital in CRAA to distinguish normal and anomalies, this loss of temporal resolution makes it exceptionally difficult to identify universally robust and distinguishable features across the spectrum of respiratory pathologies.

In conclusion, machine learning systems built upon hand-crafted features exhibit limited generalization and inferior classification performance primarily due to two fundamental limitations. First, their reliance on fixed mathematical formulations makes them inherently sensitive to real-world noise and recording artifacts,

leading to the extraction of irrelevant or misleading acoustic information. Second, their inability to effectively model the complex, non-linear, and non-stationary characteristics of adventitious lung sounds results in the loss of critical temporal and spectral details necessary for accurate diagnosis. As a consequence, models based on these traditional representations struggle to achieve the robustness and diagnostic specificity required for reliable clinical deployment [8, 26].

2.1.2 Frequency and Time Representation

Recently, researchers have emphasized on the second group where the audio recordings are transformed into two-dimensional spectrograms. These spectrograms, which can be visualized as images containing both temporal and spectral information, are then classified by deep learning based models. Inspired by this advantage, some systems utilize time-frequency representation using S-transform [22], MFCC spectrogram [46], and Log-Mel spectrogram [24] to capture concurrent variations and structural components of fundamental lung sounds. In addition, most state-of-the-art systems for detecting anomalies from respiratory sounds use Mel-filterbanks [27, 28] or short-time Fourier transformation (STFT) [17, 31, 32] to generate spectrograms. While MFCC and Log-Mel spectrograms were found as a popular technique, their reliance on fixed window sizes during FFT limits their ability to capture fine-grained spectral and temporal information [28], which is essential to represent the discrimination between normal and abnormal respiratory sounds. This fixed resolution causes an unavoidable time-frequency trade-off. For example, a window optimal for resolving short-duration, high-frequency transients (in crackles) is typically too short to accurately resolve long-duration, low-frequency events (in wheezes), and vice versa. This inflexibility severely limits the capacity to capture the fine-grained spectral and temporal features essential for highly accurate discrimination between normal and abnormal respiratory sound events. Although a variety of spectrograms have

been proposed (noting that spectrogram is conventionally exploited by a back-end deep learning model) and achieved very good performance compared with hand-crafted features, none of the publications to date have provided a comprehensive analysis of how to select the optimal kind of spectrogram, and whether there is a type of spectrogram that works well on lung sounds. To address this limitation, we propose an alternative approach based using multiple scalograms derived from Wavelet-based transformation [27, 29, 35, 43, 47]. This approach not only generates a better multi-resolution analysis thanks to its suitability in adjusting in temporal window length, but also shows great ability to capture the energy of the sound signals in wider frequency range, which is strongly relevant to the nature of respiratory sounds. In addition, we also conducted joint-learning architectures, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks to demonstrate the performances of these methods in CRAA.

In summary, while spectrogram-based representations such as MFCC, Log-Mel, and STFT have improved respiratory sound classification compared to traditional hand-crafted features, their fixed time–frequency resolution limits their ability to effectively capture both short transient events (e.g., crackles) and longer continuous patterns (e.g., wheezes). Moreover, despite their widespread adoption in deep learning frameworks, there is still no comprehensive study identifying the most suitable spectrogram type for lung sound analysis. To address these limitations, this thesis proposes a multi-resolution approach using wavelet-based scalograms, which provide adaptive temporal windowing and improved energy localization across a wider frequency range. When integrated with advanced deep learning architectures—including joint-learning, ensemble models, spatio-temporal attention mechanisms, and Inception-Residual networks—this framework offers a more robust and discriminative solution for computer-assisted respiratory anomaly analysis (CRAA).

2.2 Artificial intelligence (AI) models in CRAA

2.2.1 Traditional Machine Learning

Regarding machine learning methods [18, 19, 21], respiratory sounds are first transformed into Mel-frequency cepstral coefficient (MFCC), referred to as hand-crafted features. Then, conventional machine learning models such as Hidden Markov Model [19], Support Vector Machine [21], and Decision Trees [18] process these hand-crafted features to classify respiratory anomalies. Similarly, Taloba et al. [48] employed the Forward Feature Selection (FFS) test to statistically identify the most effective features for classification, comparing the performance contributions of MFCCs, and a set of basic time domain features (power, mean, variance, skewness, and kurtosis). In a different approach, Himadri et al. [49] utilized Linear Predictive Cepstral Coefficient (LPCC)-based features. This feature was shown to effectively outperform conventional MFCC-based techniques. The extracted respiratory audio feature vectors from these and similar systems were classified using a variety of traditional machine learning techniques [18, 19, 21, 48, 49], including Logistic Regression (LR), k-Nearest Neighbor (KNN), Hidden Markov Models (HMM), Support Vector Machines (SVM), and Decision Trees (DT). Meanwhile, respiratory sounds are transformed into a two-dimensional spectrogram, where both temporal and spectral information are capturing both temporal and spectral information over a broader time context.

2.2.2 Deep Learning

In general, three principal methodologies of deep learning are currently applied in this domain: direct training (training from scratch), transfer learning (utilizing pre-trained models), and ensemble methods.

As regards direct training of deep neural network architecture, several methods

transform the audio recordings into two-dimensional spectrograms. These spectrograms include Gammatonegram [27, 28], Mel-Frequency Cepstral Coefficients (MFCC) spectrogram [28], and Log-Mel spectrogram [29, 30]. These spectrograms are generated to capture spectral and temporal information of respiratory sounds. Next, these spectrograms are used as inputs for network architectures such as convolutional neural networks (CNNs) [29, 32, 35, 36] for classification. In detail, with method training of the model from scratch with supervised techniques, some notable studies illustrate these approaches: Acharya et al [30] proposed a hybrid CNN-RNN deep learning model to enhance classification performance by learning temporal and frequency variance. On the other hand, research in [35, 36, 50, 51] employed CNN-based architecture to analyze the temporal-frequency features embedded within the MFCC spectrograms of lung sounds, thereby facilitating the detection of pathological patterns and the inference of underlying health conditions. As a concrete demonstration, utilizing the ICBHI 2017 dataset, Yu et al [52] developed a CNN-based model that extracted both global and local features from the Log-Mel spectrogram to classify four distinct respiratory states: normal, crackle, wheeze, and combined crackle and wheeze. These studies highlight the increasing trend toward hybrid and transfer learning models, which address the complexity and variability of respiratory sound analysis.

For transfer learning from pre-trained models, this methodology can enhance the efficacy of lung sound analysis by incorporating generalized knowledge derived from extensive, related audio corpora. This capacity is potential for clinical applications, where the acquisition and curation of sufficient high-quality clinical data are frequently hindered. For example, Truc et al [37] leveraged pre-trained ResNet models as the foundation model together with various techniques such as fine-tuning, co-tuning, and stochastic normalization to outperform other classification systems for the adventitious lung sounds and respiratory diseases.

As regards ensemble methods, research in [53, 54, 55] have represented a crucial

advancement in achieving high-accuracy, robust respiratory classification by aggregating the predictions of multiple baseline models. Common strategies such as stacking, bagging, and weighted voting [54] are employed to leverage the strengths of diverse architectures (e.g., combining different classification results) or different feature sets (e.g., combining different spectrograms). By mitigating the risk of individual model error and overfitting, the collective decision-making process of an ensemble provides superior generalization capability and enhanced classification robustness, leading to significantly improved diagnostic reliability in clinical support systems [53, 55].

While techniques such as transfer learning or ensemble methods mitigate the need for massive datasets, they still rely on generalized knowledge, which may not fully capture the nuanced pathological patterns specific to lung diseases. Furthermore, the inherent complexity and noisy artifacts of real-world auscultation environments introduce significant challenges. Therefore, future improvements must focus on developing models with enhanced generalization capability to diverse, unseen clinical environments and on methods for robust feature extraction that can effectively isolate subtle respiratory anomalies from background noise and inter-patient variability. This is crucial for further research on the search for reliable and widely deployable clinical support systems.

2.3 Evaluation Metrics applied in CRAA

All provided datasets aim to focus on the detection of respiratory sound events including Normal, Wheeze, and Crackle. To adhere to the evaluation metrics in all datasets, we perform the evaluation by using Accuracy (Acc), Sensitivity (SE) and Specificity (SP) and also to derive Average score (AS), Harmonic score (HS) and Total score (TS).

We denote V is associated with the number of target classes excluding Normal

class in each task of CRAA. P_i and P_N denote the number of correctly classified samples of target class i , and class Normal, respectively. Q_i and Q_N are the numbers of all samples of target class i and class Normal, respectively. Therefore, we can establish formula for several key metrics as following:

2.3.1 Sensitivity

$$SE = \frac{\sum_{i=1}^V P_i}{\sum_{i=1}^V Q_i} \quad (1)$$

2.3.2 Specificity

$$SP = \frac{P_N}{Q_N}, \quad (2)$$

2.3.3 Average Score

$$AS = \frac{SE + SP}{2} \quad (3)$$

2.3.4 Harmonic Score

$$HS = \frac{2 \cdot SE \cdot SP}{SE + SP} \quad (4)$$

2.3.5 Score

$$TS = \frac{AS + HS}{2} \quad (5)$$

2.3.6 Accuracy

$$Acc = \frac{\sum_{i=1}^V P_i + P_N}{\sum_{i=1}^V Q_i + Q_N} \quad (6)$$

2.3.7 Summary of Evaluation Metrics in CRAA

In a classification task focused on detecting respiratory sound events (Normal, Wheeze, and Crackle), it is essential to use a suite of metrics to thoroughly evaluate model performance.

Sensitivity is known as Recall or the True Positive Rate, measures the proportion of actual positive cases that were correctly identified. In CRAA, this would be the percentage of actual Wheeze or Crackle sounds that the model correctly identified. This metric is crucial for medical applications where missing a positive case can have serious consequences. A high sensitivity means the model is good at finding all the positive instances, minimizing False Negatives. For example, it is not good to have a CRAA system to miss a crackle sound that indicates a serious condition.

Specificity is equal to the True Negative Rate, measures the proportion of actual negative cases that were correctly identified. This would be the percentage of actual Normal sounds (or non-target sounds) that the model correctly identified as Normal. This metric is important for minimizing False Positives. A high specificity means the model is good at avoiding incorrect alarms. In a medical context, this helps prevent unnecessary follow-up tests or misdiagnoses. For instance, it should be avoided in the case that the model mistakenly classifies a Normal breathing sound as a Wheeze.

Average Score is typically the mean of a set of scores. It is the average of the average of Sensitivity and Specificity applied for all tasks in CRAA. This metric provides a single, consolidated score that can be used to compare models. It attempts to give a more balanced view than just using Sensitivity or Specificity alone, but it can still be influenced by a model that performs exceptionally well on one metric while poorly on others.

Harmonic Score is analogous to the F1-Score, which is the harmonic mean of precision and recall (sensitivity). It gives equal weight to both false positives

and false negatives, making it a good measure for imbalanced datasets. This metric is particularly useful to balance Sensitivity and another metric like Precision (the proportion of positive predictions that were actually correct). It is a better overall measure than a simple average when both finding all positive cases (high sensitivity) and avoiding false alarms (high precision) are important. For example, a low F1-Score suggests that there is an imbalance between precision and recall, meaning the model is either missing many positive cases or incorrectly flagging many negative cases.

Score is Total Score based on Average Score and Harmonic Score to provide a single performance measure tailored to CRAA evaluation. In order to provide a single, final value that captures the model’s overall performance based on a specific weighting of the individual metrics, this allows for a clear ranking of models based on the specific priorities of the project (e.g., prioritizing sensitivity over specificity if a false negative is more costly than a false positive).

Accuracy provides a general sense of how well the model is performing. It is a great metric point but can be misleading, especially in datasets where some classes are much more common than others (imbalanced datasets). For example, if 95% of samples are Normal, a model that always predicts Normal would have 95% accuracy, even though it fails to detect any Wheeze or Crackle events.

2.4 Benchmark Datasets in CRAA

2.4.1 ICBHI 2017

The ICBHI dataset [56] contains 920 annotated audio samples from 126 subjects of all ages, including infants, adults, and the elderly. They are healthy individuals and patients with eight specific diseases: Chronic Obstructive Pulmonary Disease (COPD), Asthma, Lower Respiratory Tract Infection (URTI), Upper Respiratory Tract Infection (LRTI), Bronchiectasis, Heart failure, Diabetes, and Hypertension.

These audio samples were recorded by the Escola Superior de Saúde da Universidade de Aveiro (ESSUA) research team at the Respiratory Research and Rehabilitation Laboratory (Lab3R) at the Hospital Infante D. Pedro in Aveiro, at the Papanikolaou General Hospital in Thessaloniki, and at the General Hospital of Imathia (Health Unit of Naousa) in Greece. All recordings adhered to the computerized respiratory sound analysis guidelines and captured sounds from seven chest locations: trachea; left and right anterior, posterior, and lateral regions. The sounds were collected from both clinical and home environments.

The recordings were recorded using different digital stethoscopes such as the AKGC417L, Meditron, Litt3200, and LittC2SE. Subsequently, three experienced physicians (two pulmonologists and one cardiologist) annotated the recordings, segmenting them into respiratory cycles ranging from 0.2 to 16 seconds in duration, with an average length of 2.7 seconds. The dataset includes 3,642 Normal events (N), 1,864 events with Crackles (C), and 886 events with Wheezes (W), as detailed in Table 2.

2.4.2 HF-Lung-v1

This dataset [57], has 9,765 audio files of lung sounds (duration of 15 seconds each) containing 18,349 exhalation labels (as it is Normal respiratory), 8,457 wheeze labels and 15,606 discontinuous adventitious sound labels (all crackles) as detailed in Table 2. All patients were adults, over 20 years old.

Breathing lung sounds were captured at eight chest locations with two different devices: a commercial electronic stethoscope (Littmann 3200, 3M, Saint Paul, Minnesota, USA) and a customized multichannel acoustic recording device (HF-Type-1), capable of connecting eight dielectric microphones. Both devices operated at a sampling rate of 4 kHz and a bit depth of 16 bits. The recordings were saved in .wav format.

Three physicians: two respiratory therapists and one nurse, with 8, 3, and

13 years of clinical experience, respectively were recruited to identify and label specific audio events in lung sound recordings. These events included normal breathing, wheezing, and crackles. The physicians listen the audio recordings and used specialized software [58] to reconfirm on their corresponding visual representations of each event (i.e spectrograms). In order to ensure the consistency, the team established clear guidelines for labeling and regularly reviewed their work to maintain agreement on the criteria [59]. Ambiguous or incomplete sounds were excluded from labeling. Additionally, all types of crackles, regardless of severity (coarse or fine crackles), were categorized under a single C label within this dataset.

2.4.3 SPRSound 2023

We utilize the 2023 SPRSound: Open-Source SJTU Paediatric Respiratory Sound dataset, collected at the Shanghai Children’s Medical Center (SCMC) in China [60]. This dataset includes 2,683 audio recordings from 292 patients aged between 1 month and 18 years. The recordings were captured by using a Yunting model II Stethoscope with sampling rate of 8 KHz at 16-bit precision. Each recording was meticulously reviewed by experts and audio events were labelled with an onset (start time) and an offset (end time) for each audio event. Consequently, this dataset contains audio events categorized as Normal, Wheeze, Crackle (further subdivided into Coarse and Fine Crackles) as shown in Table 2.

For label annotation, a team of 11 experienced paediatric physicians participated to establish a gold standard reference for the dataset, which also makes this dataset having the largest number of annotators, compared to all other datasets [60]. This helped to reduce subjectivity and enhance the quality assurance process. High-quality signals were segmented into multiple respiratory sound events. Labels were deemed as gold standard if there was 100% agreement among physicians. In the case of having inconsistent labels, the records were transferred

to authoritative physicians for further adjudication, enabling this dataset to provide an excellent quality assurance methodology. Accordingly, this dataset served as the primary source in this research.

2.4.4 Coswara Dataset

The Second 2021 DiCOVA Challenge uses a subset of the Coswara dataset [61] collected between April 2020 and July 2021 from the age group of 15 to 90. The challenge provides a dataset of different sound signals: cough, speech, and breathing gathered from both COVID-19 positive and COVID-19 negative individuals. Given cough, speech, and breath sounds, the Second 2021 DiCOVA Challenge proposes four tracks that aim to detect COVID-19 positive subjects by exploring only breathing (Track-1), only cough (Track-2), only speech (Track-3), or all sound signals (Track-4).

We aim to focus on all tracks. While Development set is used for training, and then obtaining the best model, Blind Test set is used for evaluating and comparing the systems' performance submitted. In Development set, there are a total of 793 negative labels and 172 positive labels, which shows a significantly imbalanced dataset.

The Area under the ROC curve (AUC) is used as the primary evaluation metric in the Second 2021 DiCOVA Challenge. The curve is obtained by varying the decision threshold between 0 and 1 with a step size of 0.0001. Additionally, the Specificity and Sensitivity are used as the secondary evaluation metrics obtained at every threshold value. Note that Specificity is required to be equal or greater than 95%).

2.4.5 Dataset Conclusions

Overall, ICBHI 2017, HF-Lung-v1, SPRSound, and Coswara represent key benchmarks for classifying respiratory audio anomalies, each with distinct opportunities

and challenges.

ICBHI 2017 offers a diverse set of recordings from various sources, making it excellent for developing robust models that can handle different acoustic conditions. However, it presents a significant challenge with its imbalance, where the number of normal breath is significantly higher than abnormal ones, which can hinder the performance of classification models on rare pathologies. HF-Lung-v1 is a much larger dataset, providing diverse data for deep learning models. Its detailed annotations of breath phases and specific adventitious sounds offer a richer context for analysis. The primary challenge, however, is that all recordings were collected using the same device, which may limit the generalizability of trained models to a wider range of clinical settings and equipment. Next, SPRSound addresses a crucial gap in the field by focusing exclusively on pediatric respiratory sounds, a population with unique physiological acoustics. Its expert-annotated, high-quality data is a major opportunity for developing specialized diagnostic tools for children. The main challenge is the dataset's smaller size and the inherent noisiness of pediatric recordings, which can include sounds like crying or talking, making accurate classification more difficult. Finally, Coswara is a massive, crowdsourced open-source database of respiratory sounds designed to help with the remote diagnosis of COVID-19. It offers a significant opportunity due to its sheer size and the diversity of its recordings, which were gathered from thousands of participants across various demographics. This allows for the development of highly generalizable machine learning models. However, the dataset also presents notable challenges, primarily because its crowdsourced nature leads to significant background noise and inconsistent audio quality.

In conclusion, these datasets highlight the progression of research from general-purpose to specialized, high-fidelity data, while underscoring the ongoing challenges of class imbalance, data diversity, and clinical noise.

2.5 Summary

Despite the significant advancements in biomedical signal processing, several key challenges persist in the field of CRAA.

A major issue is the high variability of respiratory recordings, which arise from differences in patient demographics, recording environments, and the devices used. These factors introduce domain shifts that can significantly degrade the performance of classification models. In addition, there is no clear consensus on the most effective representation of respiratory sounds. While deep learning approaches commonly utilize spectrograms, a comprehensive evaluation of the most suitable type has yet to be conducted. To address this, Chapter 3 explores optimal time–frequency representations and proposes a wavelet-based multi-resolution approach.

Another critical challenge is designing a deep learning architecture that maintains robust performance under real-world noise conditions, including speech and other physiological sounds. Chapter 4 addresses this by developing and evaluating advanced deep learning architectures tailored for respiratory anomaly classification. To further enhance generalization across diverse clinical settings, Chapter 5 introduces a domain adaptation and embedding alignment framework, enabling cross-dataset robustness despite different recording conditions.

Overall, this thesis aims to overcome these challenges by developing a non-invasive, scalable, and intelligent framework for respiratory disease detection using deep learning and audio analysis. The principal impacts and contributions of this work are summarized in Chapter 6.

Table 2: Details of information on datasets used in this work.

Feature	ICBHI 2017	SPR 2023	HF-Lung-v1
Sample Count			
Normal	3642	6887	18349
Wheeze	886	865	8457
Crackle	1864	1233	15606
Recording Parameters			
Recording devices	AKGC417L, Meditron, Litt3200 and LittC2SE	Yunting model II	HF-type-1, Litt3200
Sampling rate	44.1 kHz	8 kHz	4 kHz
Bit-depth	16 bits	16 bits	16 bits
Format	.wav	.wav	.wav
Annotators (#)	3	11	3
Patient Demographics			
Participants (#)	126	292	279
Age (Median / Range)	43 (0.2 to 79.2)	5.4 (0.1 to 18)	67.5 (36.7 to 98.3)
Respiratory diseases	COPD, Asthma, Lower respiratory tract infections, Upper respiratory tract infections, Bronchiecta- sis, Heart failure, Diabetes, Hypertension	Asthma, Bronchitis, Pneumonia (Non-severe), Pneumonia (Severe)	Acute failure, Chronic failure, Emphysema, Pneumonia, Acute distress syndrome, COPD, Acute exacerbation of chronic

Chapter 3

Audio Feature Extraction

This chapter presents a comprehensive introduction of various spectrograms and scalograms representations used for robust acoustic modeling and respiratory audio feature extraction. The objective is to establish a significant foundation that can enable a crucial evaluation of different proposed CRAA frameworks detailed in Chapter 4.

3.1 Short Time Fourier Transform

Short-Time Fourier Transform (STFT): The STFT spectrogram evaluated applies Fourier Transform to extract Frequency content of local section of input signal over short time duration [62, 63]. Let consider $\mathbf{s}(n)$ as digital audio signal with length of N , a pixel value at central frequency F and time frame T of STFT spectrogram $\mathbf{STFT}[F, T]$ is computed as:

$$\mathbf{STFT}[F, T] = \sum_{n=0}^{N-1} \mathbf{s}[n + Th] \cdot \mathbf{w}[n] e^{-j \frac{2\pi kn}{N}} \quad (7)$$

where $\mathbf{w}[n]$ is a window function, typically Hamming [64]. h is the hop size. While time resolution (T) of STFT spectrogram determined using window side and

hope size, the frequency resolution (F) equals to the number of central frequencies set to 2048. The frequency resolution, eventually, is rescaled to 128 to match the resolution of other spectrograms.

3.1.1 Log-Mel

log-Mel: To generate log-Mel spectrogram, draw audio signal is firstly transformed into STFT spectrogram recently mentioned. Next, a Mel filter bank, which simulates the overall frequency selectivity of the human auditory system using the frequency warping $F_{mel} = 2595 \cdot \log(1 + F/700)$ [65], is applied to generate a Mel spectrogram $\mathbf{MEL}[F_{mel}, T]$ (noting that frequency resolution (F_{mel}) is the Mel filter number set to 128). Eventually, logarithmic scaling is applied to obtain the log-Mel spectrogram. Let consider $\mathbf{COE}[F_{mel}, F]$ as matrix storing coefficients of Mel filters, log-Mel spectrogram likely a matrix is computed by:

$$\mathbf{log-Mel}[F_{mel}, T] = \log \mathbf{COE}[F_{mel}, F] \cdot |\mathbf{STFT}[F, T]| \quad (8)$$

3.1.2 MFCC

Mel Frequency Cepstral Coefficient (MFCC): From log-Mel spectrogram, Discrete Cosine Transform (DCT) is used to extract a sequence of uncorrelated coefficients crossing frequency dimension, reducing log-Mel frequency resolution into smaller space [66]. A pixel value $\mathbf{DCT}[f_{dct}, t_{dct}]$ of DCT matrix $\mathbf{DCT}[F_{dct}, T_{dct}]$, where F_{dct} and T_{dct} are frequency and time resolutions, is computed by:

$$\begin{aligned}
\mathbf{DCT}[f_{dct}, t_{dct}] &= \left(\frac{2}{F_{mel}}\right)^{\frac{1}{2}} \cdot \left(\frac{2}{T}\right)^{\frac{1}{2}} \cdot \sum_{f_{mel}=0}^{F_{mel}-1} \cdot \sum_{t=0}^{T-1} \\
&\quad \Lambda(f_{mel}) \cdot \cos \left[\frac{\pi f_{dct}}{F_{mel}} (2f_{mel} + 1) \right] \\
&\quad \cdot \Lambda(t) \cdot \cos \left[\frac{\pi t_{dct}}{T} (2t + 1) \right] \\
&\quad \cdot \mathbf{log-Mel}[f_{mel}, t]
\end{aligned} \tag{9}$$

where

$$\Lambda(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{if } x = 0 \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

T and F_{mel} are time and frequency resolution of log-Mel spectrogram.

Next, delta coefficients per time frame showing difference of DCT coefficients over time are computed, shown in Equation. (11).

$$\begin{aligned}
\mathbf{DELTA}[F_{dct}, t] &= \frac{1}{2} (\mathbf{DCT}[F_{dct}, t - 1] \\
&\quad - \mathbf{DCT}[F_{dct}, t + 1])
\end{aligned} \tag{11}$$

$\mathbf{DELTA}[F_{dct}, T_{dct}]$ is concatenated with DCT spectrogram. $\mathbf{DCT}[F_{dct}, T_{dct}]$ across frequency dimension to generate MFCC spectrogram $\mathbf{MFCC}[F_{mfcc}, T_{dct}]$ (noting that MFCC frequency resolution (F_{mfcc}) doubles frequency resolution of DCT (F_{dct}) and equals to 128, and T_{dct} is set to equal to T resolution of log-Mel spectrogram).

3.1.3 CQT

Constant Q transform (CQT): This spectrogram applies a bank of filters corresponding to tonal spacing [67], where each filter is equivalent to a subdivision

of an octave, with central frequencies given by:

$$f_k = (2^{\frac{1}{b}})^k \cdot f_{min} \quad \text{for } 1 \leq k \leq K \quad (12)$$

where f_k denotes the frequency of k th spectral component, f_{min} is minimum frequency set to 10 Hz, b is the number of filters per octave as 24, and K is frequency resolution of CQT, which is 128. As the name suggest, the Q value, is the ratio of central frequency to bandwidth, is constant computed as:

$$Q = \frac{f_k}{\Delta f_k} = \frac{f_k}{f_{k+1} - f_k} = \left(2^{\frac{1}{b}} - 1\right)^{-1} \quad (13)$$

Like STFT, CQT spectrogram is extracted using Fourier-based transformation, described as Equation (14) :

$$\mathbf{CQT}[f_k, t] = \frac{1}{N(k)} \cdot \sum_{n=0}^{N(k)-1} \mathbf{s}[n] \cdot \mathbf{w}[k, n - t] \cdot e^{-i2\pi \frac{nQ}{N(k)}} \quad (14)$$

where:

$$N(k) = Q \cdot \frac{f_s}{f_k} \quad (15)$$

$$\mathbf{w}[k, n] = \alpha + (1 - \alpha) \cdot \cos \frac{2\pi n}{N(k) - 1} \quad (16)$$

f_s is sampling rate of digital input signal $\mathbf{s}[n]$, $\mathbf{w}[k, n]$ is window function with α set to 0.54. In this work, STFT, log-Mel, MFCC, and CQT features were generated using the Librosa audio processing library [66].

3.1.4 Gammatone

Gammatone (GAM): Gammatone filters are designed to model the frequency-selective cochlea activation response of the human inner ear [68], in which filter

output simulates the frequency response of the basilar membrane. The impulse response is given by:

$$g(t) = t^{P-1}e^{-2lt\pi}\cos(2ft\pi + \theta) \quad (17)$$

where t is time, P is the filter order, θ is the phase of the carrier, l is filter bandwidth, and f is central frequency. The filter bank was then formulated as ERB scale [69] as:

$$ERB = 24.7(4.37 \cdot 10^{-3}f + 1) \quad (18)$$

To quickly generate Gamma spectrogram, we apply a toolbox developed by Ellis *et al.* [70], namely Gammatone-like spectrogram. Firstly, audio signal is transformed into STFT spectra as described earlier. Next, gammatone weighting $\mathbf{COE}[F_{gam}, F]$ is applied on STFT to obtain the Gamma spectrogram.

$$\mathbf{GAM}[F_{gam}, T] = \mathbf{COE}[F_{gam}, F] \cdot |\mathbf{STFT}[F, T]| \quad (19)$$

where F_{gam} resolution of GAM spectrogram is Gammatone filter number of 128.

3.2 Foundational approach on Multi-spectrogram Features and Joint Learning Models

We begin in this Section 3.2 by conducting extensive acoustic signal processing experiments on spectrograms. This preliminary work focuses on acoustic scene classification and evaluates diverse acoustic modeling methods, establishing a solid foundation for CRAA. The study then offers comprehensive insights into respiratory audio feature extraction, which is critical for analyzing advanced deep learning architectures. Specifically, we explore how joint-learning, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks can effectively capture spatial and temporal information in respiratory

audio.

Our proposed deep learning framework applied for Acoustic Scene Classification (ASC), in general, is described in Figure 2. The framework is separated into low-level feature extraction (the upper part) and back-end classification (the lower part). In particular, the draw audio from the Channel 1 is firstly transformed into spectrogram representation, using 128 filter banks. The spectrogram is divided into non-overlapping patches of size 128×128 .

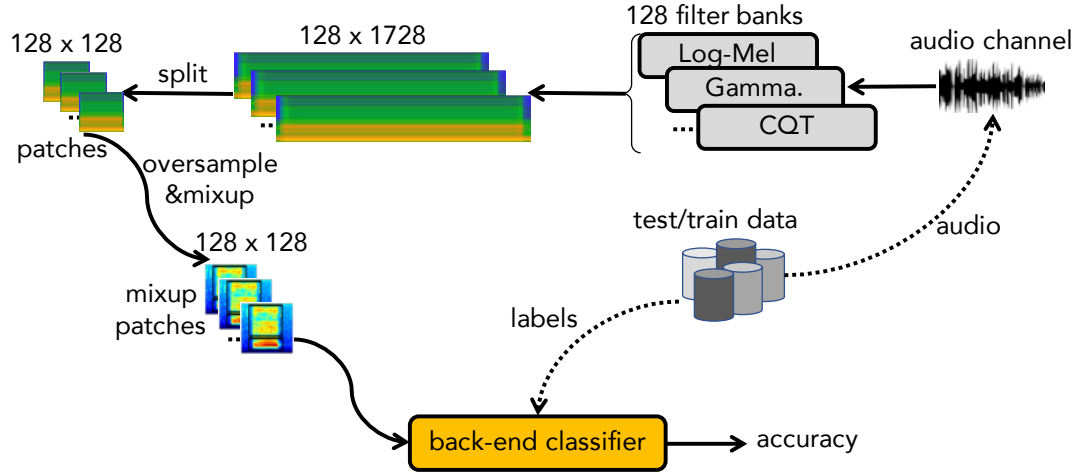


Figure 2: High-level architecture of our Acoustic Scene Classification system.

To deal with unbalanced data issue, we apply two data augmentation techniques on the image patches. Firstly, we randomly oversample patches from underrepresented categories. Next, the mixup data augmentation [71] is applied, thus generate new image patches. Let's consider two original image patches as $\mathbf{X}_1$, $\mathbf{X}_2$ and expected labels as $\mathbf{y}_1$, $\mathbf{y}_2$, new image patches are generated as below equations:

$$\mathbf{X}_{\text{mp1}} = \mathbf{X}_1 * \gamma + \mathbf{X}_2 * (1 - \gamma) \quad (20)$$

$$\mathbf{X}_{\text{mp2}} = \mathbf{X}_1 * (1 - \gamma) + \mathbf{X}_2 * \gamma \quad (21)$$

$$\mathbf{y}_{\text{mp1}} = \mathbf{y}_1 * \gamma + \mathbf{y}_2 * (1 - \gamma) \quad (22)$$

$$\mathbf{y}_{\text{mp2}} = \mathbf{y}_1 * (1 - \gamma) + \mathbf{y}_2 * \gamma \quad (23)$$

where γ is random coefficient from either *unit* or *beta* distribution, $\mathbf{X}_{\text{mp1}}$, $\mathbf{X}_{\text{mp2}}$ and $\mathbf{y}_{\text{mp1}}$, $\mathbf{y}_{\text{mp2}}$ are new image patches and labels generated, respectively. Eventually, the mixup patches are fed into a back-end classifier to produce the final classification output.

3.2.1 Acoustic Datasets and Setting

Our experiments are conducted over a variety of published ASC datasets, comprising LITIS Rouen [72] and IEEE AASP Challenge on Detection and Classification of Acoustic Scenes and Events (DCASE) 2016 Task 1 [73], 2017 Task 1 [74], 2018 Task 1A & 1B [75].

LITIS Rouen dataset was recorded at a sample rate of 22050 Hz with 3026 segments, each presents 30-s duration. This dataset contains a total of 25.51 recording hours for 19 urban scene categories, showing unbalanced data. Following the mandated settings, the dataset is separated and organised for 20-fold cross validation, reporting the final classification accuracy by averaging over the 20 testing folds.

DCASE includes:

- DCASE 2016 Task 1 and DCASE 2017 Task 1 similarly present 15 categories and were recorded at 44.1 kHz. While each segment in DCASE 2016 is 30s, 10-s duration is presented in DCASE 2017. Notably, DCASE 2017 reuses all DCASE 2016 and adds new data recorded. By following the recommended setting, we train our proposed system on development set (Dev.) and evaluate on the evaluation set (Eva.).
- DCASE 2018 Task 1A, it was recorded at 44.8 kHz, spanning 10 categories and used one recording device namely A.

- DCASE 2018 Task 1B reuses all data from DCASE 2018 Task 1A, and adds more data recorded by two different devices, namely B and C. Noticeably, the total recording time conducted on device B and C are much less than device A (denoted as DCASE 2018 Task 1A), reporting totally 4 hours on B&C compared to 24 hours in device A. As a result, DCASE 2018 Task 1B dataset involves issues of mismatched recording devices and unbalanced data in terms of recording devices. Therefore, DCASE 2018 Task 1B challenge only compare systems' results on device B&C with less recording time.

As DCASE 2018 Task 1A and 1B have not released labels of evaluation set, we separate development set into two sub-sets, namely Training and Test sets for training and testing processes respectively. While DCASE 2016 Task 1 and DCASE 2017 Task 1 are balanced, a slight imbalance is present in DCASE 2018 Task 1A and 1B.

3.2.2 High-Level Architecture and Baseline C-DNN Network Proposed

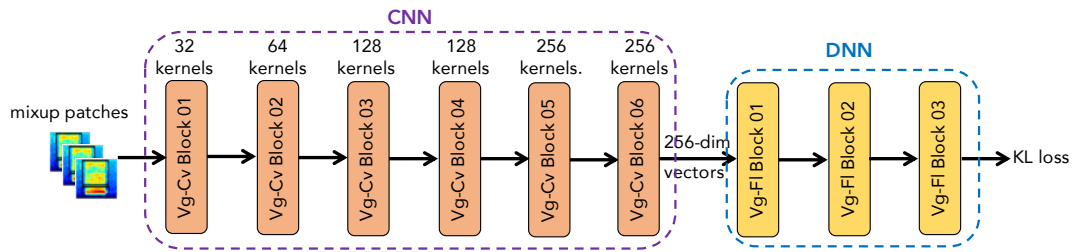


Figure 3: Block-level architecture of C-DNN network.

In order to evaluate individual and multiple spectrograms, we proposed a C-DNN network architecture as described in Figure 3 and Table 3. As shown in Figure 3, C-DNN architecture comprises of CNN and DNN parts in order. CNN component consists of 6 Vg-Cv blocks, performed by Batchnorm (Bn), Convolutional (Cv[kernel size]), Rectified linear unit (ReLu), Dropout (Dr(Percentage

Table 3: Network layers used in C-DNN architecture.

	Blocks	Layers	Output shape
CNN Architecture	Input		$128 \times 128 \times 1$
	Vg-Cv Block 01	Bn - Cv $[9 \times 9]$ - ReLu - Bn - Ap $[2 \times 2]$ - Dr (10%)	$64 \times 64 \times 32$
	Vg-Cv Block 02	Cv $[7 \times 7]$ - ReLu - Bn - Ap $[2 \times 2]$ - Dr (15%)	$32 \times 32 \times 64$
	Vg-Cv Block 03	Cv $[5 \times 5]$ - ReLu - Bn - Dr (20%)	$32 \times 32 \times 128$
	Vg-Cv Block 04	Cv $[5 \times 5]$ - ReLu - Bn - Ap $[2 \times 2]$ - Dr (20%)	$16 \times 16 \times 128$
	Vg-Cv Block 05	Cv $[3 \times 3]$ - ReLu - Bn - Dr (25%)	$16 \times 16 \times 256$
	Vg-Cv Block 06	Cv $[3 \times 3]$ - ReLu - Bn - Gap - Dr (25%)	256
DNN Architecture	Vg-Fl Block 01	Fl - ReLu - Dr (30%)	512
	Vg-Fl Block 02	Fl - ReLu - Dr (30%)	1024
	Vg-Fl Block 03	Fl - Softmax	10

dropped)), Average Pooling (Ap), Global Average Pooling (Gap) layers as shown in the top of Table 3. Meanwhile, DNN part in Figure 3 is configured by three Vg-Fl blocks with Fully-connect (Fl), ReLu, Dropout (Dr(Percentage dropped)), and Softmax layers, as described in the bottom of Table 3. It can be seen that CNN part helps to map input image patches to condensed and discriminative vectors, referred to as high-level features. Each high-level feature vector has 256 dimensions, determined by the number of kernels in the final convolutional layer of Vg-Cv block 06.

Next, DNN part explores the high-level features, thus classifies into 10 categories (the category number in DCASE 2018 Task 1B dataset evaluated) and reports the classification accuracy.

3.2.3 An Analysis of Spectrograms

By using the high-level architecture mentioned above, we evaluate five individual spectrograms (STFT, log-Mel, MFCC, GAM, and CQT), thus indicate which kind of spectrograms and their combinations is the most influencing on our system's performance. To evaluate the individual spectrograms, we firstly proposed a C-DNN based network referred to as the back-end classifier.

Next, we create five systems, each reuse the high-level framework architecture

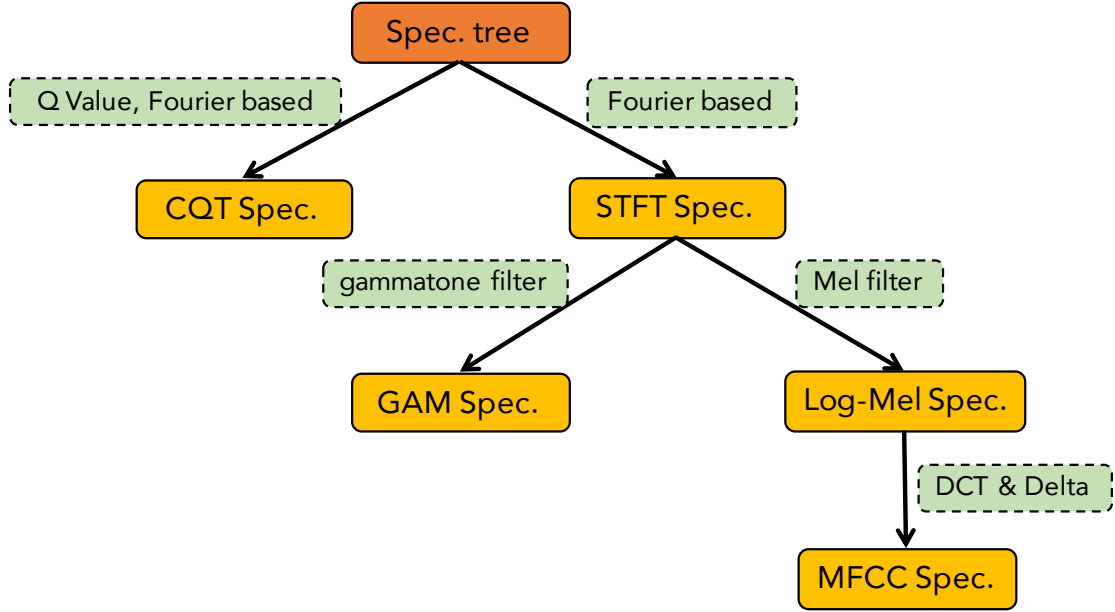


Figure 4: Constructed spectrogram tree basing on difference of central frequencies and auditory models applied.

mentioned with one type of spectrogram and C-DNN architecture for classification. Noting that we use same setting with window size=1290, hop size=256, frequency minimum $f_{min}=10$ Hz, and filter bank number=128 to generate same-size spectrograms. The results of five systems over DCASE 2018 Task 1B dataset are analysed and compare to the DCASE baseline. Inspire that each spectrogram contain discriminative and complementary features, we fuse the individual systems' accuracy results, thus indicate which combination of spectrograms is effective to improve the performance.

To propose spectrogram combinations, formulas of individual spectrograms are firstly presented in section 3.1. Based on the tree shown in Figure 4, we propose a variety of combinations as denoted in Table 4.

As spectrogram formulas described, we construct a spectrogram tree as shown in Figure 4. Firstly, although both of CQT and STFT spectrograms are built on Fourier Transform theory, they extract different central frequencies. From the root tree, we therefore separate into two main branches of CQT and STFT. From

Table 4: Spectrogram combinations proposed.

Group of	Combinations of
Two spectrograms	CQT+STFT, CQT+GAM, CQT+log-Mel, CQT+MFCC
Three spectrograms	CQT+log-Mel+GAM, CQT+GAM+MFCC
Four spectrograms	CQT+GAM+STFT+MFCC, CQT+GAM+STFT+log-Mel
Five spectrograms	CQT+log-Mel+GAM+STFT+MFCC

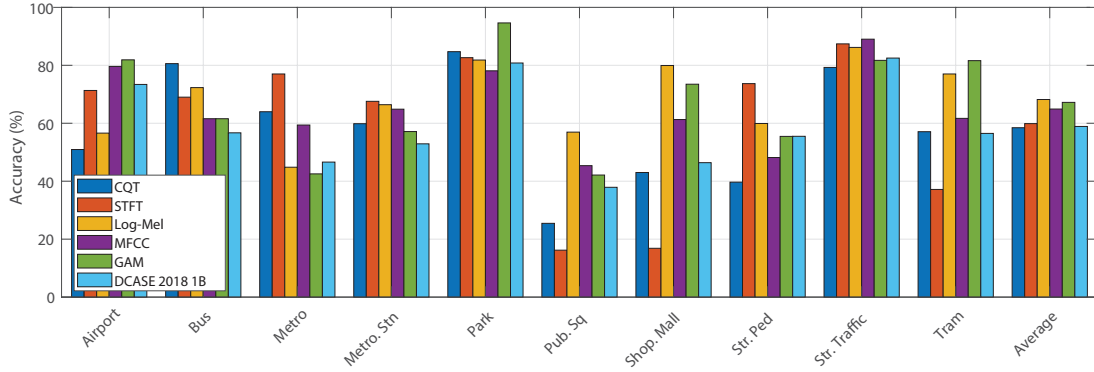


Figure 5: Category-wise performance comparison among spectrograms on device A - DCASE 2018 Task 1B.

the branch of STFT spectrogram, we continuously divide into log-Mel and GAM spectrograms due to applying different Mel and Gammatone filters, respectively. Eventually, MFCC is an extended branch from log-Mel due to extracting DCT and Delta from this spectrogram. It can be seen that five spectrograms proposed either extract different central frequencies or apply different auditory models. Therefore, each spectrogram may contain its own distinct and complementary information. This inspires us to conduct experiments to evaluate how individual spectrograms and their combinations affect to ASC system's performance. Based on the tree shown in Figure 4, we propose a variety of combinations as denoted in Table 4. In particular, two-spectrogram combinations are inspired from two main branches from the root tree, each extracts specific central frequencies. Thus, we create groups of CQT+STFT, CQT+GAM, CQT+log-Mel,

and CQT+MFCC. There are two third-spectrogram groups of CQT+GAM+log-Mel and CQT+GAM+MFCC evaluated, which inspires from exploring different central frequencies between CQT & STFT branches and different auditory models used among MFCC, log-Mel, and GAM. Since applying auditory models on STFT may distort its discriminative features, and MFCC and log-Mel share similar features due to both being derived from Mel filter banks, we propose two four-spectrogram combinations, which are CQT+GAM+STFT+MFCC and CQT+GAM+STFT+log-Mel. Eventually, the combination of all five spectrograms is also evaluated.

3.2.4 Late Fusion Strategy to Evaluate Spectrogram Combinations

As the back-end classification works on smaller patches, the posterior probability of an entire spectrogram is computed by averaging of all patches' posterior probabilities. Let us consider $\mathbf{P}^n = (\mathbf{p}_1^n, \mathbf{p}_2^n, \dots, \mathbf{p}_C^n)$, with C being the category number and the n^{th} out of N patches fed into learning model, as the probability of a test sound instance, then the mean classification probability is denoted as $\bar{\mathbf{p}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C)$ where,

$$\bar{p}_c = \frac{1}{N} \sum_{n=1}^N p_c^n \quad for \quad 1 \leq n \leq N \quad (24)$$

and the predicted label $\hat{y}$ for an individual spectrogram evaluated is determined using:

$$\hat{y} = \operatorname{argmax}(\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C) \quad (25)$$

To evaluate the combinations of spectrograms, we proposed a late fusion scheme, namely *Mean* fusion. In particular, we conduct experiments over individual spectrograms, thus obtain posterior probability of each spectrogram as

$\bar{\mathbf{p}}_s = (\bar{p}_{s1}, \bar{p}_{s2}, \dots, \bar{p}_{sC})$ where C is the category number and the s^{th} out of S spectrograms evaluated. Next, the posterior probability after late fusion $\mathbf{p}_{f\text{-mean}} = (p_1, p_2, \dots, p_C)$ is obtained from by:

$$p_c = \frac{1}{S} \sum_{s=1}^S \bar{p}_{sc} \quad \text{for } 1 \leq s \leq S \quad (26)$$

Eventually, the predicted label $\hat{y}$ is determined by:

$$\hat{y} = \text{argmax}(p_1, p_2, \dots, p_C) \quad (27)$$

In this work, we adopt Tensorflow framework to build deep learning models with learning rate of 0.0001, a batch size of 50, epoch number of 100, and Adam method [76] for learning rate optimization. As using mixup data augmentation, the labels are not one-hot format. Therefore, we use Kullback-Leibler (KL) divergence loss [77] instead of the standard cross-entropy loss as shown in Equation below:

$$Loss_{KL}(\theta) = \sum_{n=1}^N \mathbf{y}_n \log\left(\frac{\mathbf{y}_n}{\hat{\mathbf{y}}_n}\right) + \frac{\lambda}{2} \|\theta\|_2^2, \quad (28)$$

where $Loss_{KL}(\theta)$ is KL-loss function, θ describes the trainable parameters of the network trained, λ denote the ℓ_2 -norm regularization coefficient experimentally set to 0.0001, N is the batch size, $\mathbf{y}_c$ and $\hat{\mathbf{y}}_c$ are the ground-truth and the network recognized output, respectively. Noting that we use only DCASE 2018 Task 1B dataset to analyse individual spectrograms and their combinations proposed.

Initially, we evaluate all five individual spectrograms using the C-DNN network, and report category-wise performance for devices A and B&C in the DCASE 2018 Task 1B dataset. As shown in Figure 5, results on device A indicate that log-Mel, GAM, and MFCC outperform STFT and CQT in most categories.

In terms of average accuracy, log-Mel and GAM stand on the top, and achieve competitive results of 68.2% and 67.2%, respectively. Meanwhile, STFT and

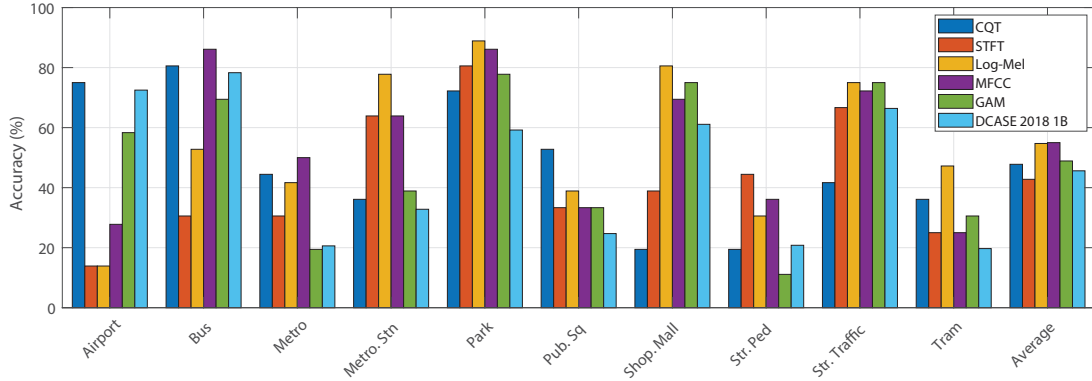


Figure 6: Category-wise performance comparison among spectrograms on device B&C - DCASE 2018 Task 1B.

CQT show very low average scores compared with log-Mel and GAM, indicating a performance gap of nearly 10%. Noticeably, CQT spectrogram showed great performance in sound scenes belong to transportation such as *Bus*, *Metro* and *Street Traffic*.

Regarding performance on device B&C in Figure 6, results show a similar trend with top scores of log-Mel, GAM, and MFCC. Again, CQT spectrogram still shows good accuracy rate in transformation-related categories such as *Bus*, *Metro* and *Tram*.

In general, category-wise performance comparison of device A and device B&C indicate that spectrograms extracted from auditory models such as GAM, log-Mel and MFCC gain high performance. Comparing these three spectrograms to the DCASE 2018 baseline in the context of Task 1B challenge (only device B&C), they outperform the DCASE 2018 baseline over almost categories and show an improvement of 3.3%, 9.1%, and 9.4% respectively in terms of the average result.

Next, we conduct experiments on two-spectrogram combinations and present obtained results in Table 5. As shown in Table 5, CQT+log-Mel achieves the greatest performance on both device A and B&C, improving by 4% and 6% respectively compared to only log-Mel (the top score of individual spectrogram).

Compare CQT+log-Mel score to DCASE 2018 Task 1B baseline, it shows a significant improvement of 13.1% and 15.2% over device A and B&C, respectively.

Table 5: Compare individual spectrograms and their combinations to DCASE 2018 Task 1B baseline with best results (%) in **bold**.

Spectrograms	A	B&C	A&B&C
<i>DCASE baseline</i>	58.9	45.6	52.2
<i>Single spectrogram</i>			
MFCC	64.9	55.0	59.9
STFT	59.8	42.7	51.3
log-Mel	68.2	54.7	61.4
CQT	58.4	47.8	53.1
GAM	64.1	48.9	58.1
<i>Two spectrograms</i>			
CQT+STFT	64.2	55.8	60.0
CQT+GAM	70.9	53.3	62.1
CQT+log-Mel	72.0	60.8	66.4
CQT+MFCC	69.8	58.9	64.4
<i>Three spectrograms</i>			
CQT+GAM+log-Mel	74.1	62.5	68.3
CQT+GAM+MFCC	71.9	61.1	66.5
<i>Four spectrograms</i>			
CQT+GAM+STFT+log-Mel	74.4	62.5	68.5
CQT+GAM+STFT+MFCC	72.7	60.3	66.5
<i>All five spectrograms</i>			
CQT+GAM+STFT+log-Mel+MFCC	73.7	62.8	68.2

As regards three-spectrogram combinations as shown in Table 5, two groups of CQT+GAM+MFCC and CQT+log-Mel+GAM achieve competitive results, reporting 71.9%, 61.1% and 74.1%, 62.5% for device A and B&C, respectively. It indicates that log-Mel and MFCC may contain very similar features.

The results on four-spectrogram combinations of CQT+GAM+STFT+log-Mel witness a minor increase of nearly 0.3% and 0.2% in terms of device A and overall respectively, compared to CQT+log-Mel+GAM, resulting from the addition of STFT. Meanwhile, adding STFT into CQT+GAM+MFCC only helps to

improve the performance on device A a little, but makes a decrease of 0.8% on device B&C. As regards the result of all five spectrograms, it even has a downward trend in device A, leading to the a decrease of 0.7 %, compared to overall accuracy in the best four-spectrogram combination of CQT+GAM+STFT+log-Mel.

Eventually, we summary all types of spectrogram combinations, and highlight which achieve the best scores on all three devices A&B&C. As results shown in Table 5, there is a gradual increase in the accuracy rate when combination of spectrograms are applied. In particular, CQT+log-Mel achieves the best performance in two-spectrogram groups, with an increase of nearly 5% compared to the best single spectrogram log-Mel. By adding GAM into group of CQT+log-Mel, it helps to improve by 2% on average. However, a minor increase of 0.2% is observed in the performance of four-spectrogram combination CQT+GAM+STFT+log-Mel until there is no improvement from combination of all five spectrograms. It can be concluded that using multiple spectrograms is effective to improve the performance, thus far exceed the DCASE 2018 Task 1B baseline.

3.2.5 Joint Learning Deep Neural Network Architecture Proposed

Table 6: C-RNN network architecture.

Blocks	Layers	Output shape
	Input	$128 \times 128 \times 1$
Re-Cv Block 01	Bn - Cv $[4 \times 1]$ - ReLu - Bn - Ap $[2 \times 1]$ - Dr (10%)	$64 \times 128 \times 32$
Re-Cv Block 02	Cv $[4 \times 1]$ - ReLu - Bn - Ap $[2 \times 1]$ - Dr (15%)	$32 \times 128 \times 64$
Re-Cv Block 03	Cv $[4 \times 1]$ - ReLu - Bn - Ap $[2 \times 1]$ - Dr (20%)	$16 \times 128 \times 128$
Re-Cv Block 03	Cv $[4 \times 1]$ - ReLu - Bn - Ap $[16 \times 1]$ - Dr (20%)	128×256
Re-Bi-GRU Block	Bi-GRU (128 hidden states, 30% dropout)	128×256
Re-GlAv Block	Gap	128

As the analysis results on spectrogram, it can be seen that using multi-spectrogram input features is effective to improve the ASC system's performance. To further enhance, we next make effort to improve the back-end classification,

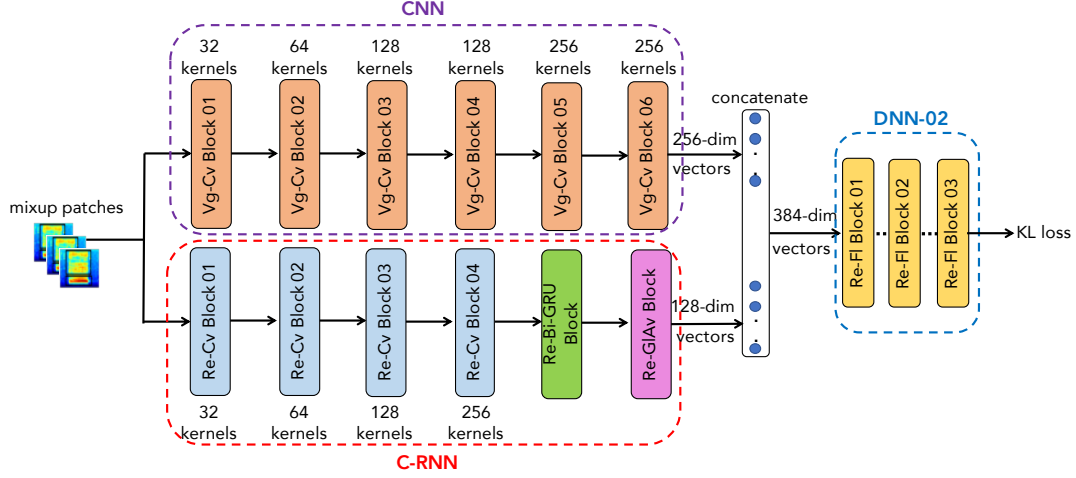


Figure 7: Joint learning network architecture.

thus propose a joint learning model, as described in Figure 7. As Figure 7 shown, we reuse the CNN part with six Vg-Cv blocks from the C-DNN architecture. These Convolutional blocks help to capture spatial features from spectrogram input, thus transform image patches into condensed and discriminative vectors with 256 dimension. Additionally, we add a parallel C-RNN architecture (the lower part of Figure 7) that is used to capture structures of temporal sequences from spectrogram input. As Table 6 showing the C-RNN proposed, input patches of 128×128 are fed into sub-blocks Cv, Bn, ReLu, Ap and Dr that are similar to those used in the CNN part. However, we adjust settings of these sub-blocks to allow the C-RNN network be able to learn time-sequential features. In particular, convolutional layers (Cv) with kernel size set to $[4 \times 1]$ are applied to learn the difference between frequency banks in each temporal frame. Next, average pooling layers (Ap $[2/16 \times 1]$) is used to scale the frequency dimension of the spectrogram, but maintaining a temporal resolution of 128. As a result, frequency dimension is scaled into 1, generating a sequence of 128-temporal frames after four Re-Cv blocks. Each temporal frame is represented by a 256-dimensional vector. Next the temporal sequence is fed into bi-GRU layer in Re-Bi-GRU Block which learns the temporal sequence structure from two directions. The output of Re-Bi-GRU

Block is a matrix of 128×256 with 128 temporal frames and 256 dimension each frame. Next, a Global Average Pooling layer in Re-GlAv block is applied on each temporal frame to get average results, generating a 128-dimensional vectors. Both output of C-RNN and CNN are thus concatenated, generate 384-dimensional vectors. Next, these vectors are fed into a DNN-02 architecture, as shown in Table 7, configured by Fl, ReLu, Dr, and Softmax layers for classification. Noting that output layer number C depends on specific ASC task due to various datasets evaluated.

Table 7: DNN-02 network architecture.

Blocks	Layers	Output shape
	Input	384
Re-Fl Block 01	Fl - ReLu - Dr (30%)	2048
Re-Fl Block 02	Fl - ReLu - Dr (30%)	1024
Re-Fl Block 03	Fl - Softmax	C

The joint learning model proposed is built by Tensorflow framework and reused all hyper-parameter setting from C-DNN network experiments. To evaluate the effect of spectrograms, we conduct experiments on the best spectrogram combinations indicated in Table 5 in Section 3.2.4.

Additionally, we do further investigation of late fusion on accuracy. In particular, we compute more two fusion strategies, called *Max* and *Prod* fusions. Let us consider posterior probability of each spectrogram describes as $\bar{\mathbf{p}}_{\mathbf{s}} = (\bar{p}_{s1}, \bar{p}_{s2}, \dots, \bar{p}_{sC})$ described in Equation (26), where s is specific spectrogram and C is the number of category classified. Next, the posterior probability of combination with *Prod* strategy $\mathbf{p}_{\mathbf{f-prod}} = (p_1, p_2, \dots, p_C)$ is obtained by,

$$p_c = \frac{1}{S} \prod_{s=1}^S \bar{p}_{sc} \quad \text{for } 1 \leq s \leq S \quad (29)$$

where S is the number of spectrograms combined. The posterior probability of

combination with *MAX* strategy $\mathbf{p_{f-max}} = (p_1, p_2, \dots, p_C)$ is obtained by,

$$p_c = \max(\bar{p}_{1c}, \bar{p}_{2c}, \dots, \bar{p}_{Sc}) \quad (30)$$

Eventually, the predicted label $\hat{y}$ for either *Max* or *Prod* fusions is determined by Equation (25).

As regards ASC datasets evaluated, we conduct extensive experiments on five different datasets, comprising of Litis Rouen, DCASE 2016 Task 1A, DCASE 2017 Task 1A, and DCASE 2018 Task 1A and 1B. Thus, we compare our best results to the state-of-the-art systems.

Performance comparison on DCASE 2018 Task 1B dataset

As results obtained in Table 8, adding the C-RNN architecture into C-DNN network to create the joint learning model helps to improve the performance over both device A and B&C. Specially, the accuracy rate increases when more spectrograms are combined.

As regards late fusion methods suggested, *Prod* and *Mean* are very competitive and outperform *Max* fusion scheme in both C-DNN and joint learning model proposed. Noticeably, joint learning models with *Prod* fusion achieve the best scores for all kinds of spectrogram combinations in terms of B&C performance.

Compare to DCASE 2018 Task 1B baseline, joint learning models proposed outperform DCASE baseline on both device A and B&C. In particular, the best score of 67.5% over devices B&C, which is obtained from combination of all spectrograms CQT+log-Mel+GAM+STFT+MFCC, significantly improves the DCASE baseline by 22%. It indicates that the strategy of multi-spectrogram input successfully solve the problem of mismatched devices A and B&C raised in DCASE 2018 Task 1B challenge.

Performance comparison to the-state-of-the-art systems

We continue to evaluate our best proposed systems on various datasets, thus make a comparison to the state-of-the-art at time of writing. As detail presented

Table 8: Performance comparison (%) on DCASE 2018 Task 1B dataset with the highest scores highlighted in **bold**.

Architecture	Joint Learning Model			C-DNN		
Device A	Mean	Prod	Max	Mean	Prod	Max
CQT+log-Mel	72.4	72.1	70.9	72.0	73.0	70.1
CQT+log-Mel+GAM	74.9	74.5	73.4	74.1	74.7	72.3
CQT+log-Mel+GAM+STFT	76.2	76.5	73.6	74.4	74.9	71.5
CQT+log-Mel+GAM+STFT+MFCC	76.0	76.4	74.0	73.7	74.6	71.5
Devices B & C:	Mean	Prod	Max	Mean	Prod	Max
CQT+log-Mel	62.2	64.7	60.3	60.8	62.2	59.2
CQT+log-Mel+GAM	65.00	66.4	61.9	62.5	63.3	59.2
CQT+log-Mel+GAM+STFT	64.44	66.7	60.6	62.5	60.6	58.3
CQT+log-Mel+GAM+STFT+MFCC	65.3	67.5	64.7	62.8	61.9	58.1

Table 9: Comparison of the state-of-the-art systems with best performance highlighted in **bold**.

D. 2018 Task 1B (Dev. set)	Acc. (%)	D. 2018 Task 1A (Dev. set)	Acc. (%)	D. 2017 Task 1 (Eva. set)	Acc. (%)	D. 2016 Task 1 (Eva. set)	Acc. (%)	LITIS Roune (20-fold Ave.)	Acc. (%)
Baseline [75]	45.6	Baseline [75]	59.7	Baseline [73]	74.8	Baseline [78]	77.2		
Li [79]	51.7	Li [80]	72.9	Zhao [81]	70.0	Wei [82]	84.1	Bisot [83]	93.4
Tchorz [84]	53.9	Jung [85]	73.5	Jung [86]	70.6	Bae [87]	84.1	Ye [88]	96.0
Kong [89]	57.5	Wang [90]	73.6	Karol [91]	70.6	Kim [92]	85.4	Huy [93]	96.4
Wang [94]	57.5	Christian [95]	74.7	Ivan [96]	71.7	Takahasi [97]	85.6	Yin [98]	96.4
Waldekar [99]	57.8	Zhang [100]	75.3	Park [101]	72.6	Elizalde [102]	85.9	Huy [103]	96.6
Zhao [104]	63.3	Li [79]	76.1	Lehner [105]	73.8	Valenti [106]	86.2	Ye [107]	97.1
Truc [108]	63.6	Dang [109]	76.7	Hyder [110]	74.1	Marchi [111]	86.4	Huy [112]	97.8
		Octave [113]	78.4	Zhengh [114]	77.7	Park [115]	87.2	Zhang [116]	97.9
		Yang [117]	79.8	Han [118]	80.4	Bisot [119]	87.7	Zhang [120]	98.1
		Golubkov [121]	80.1	Mun [122]	83.3	Hamid [123]	89.7	Huy [124]	98.7
Zhao [104]	63.3	Bai [125]	66.1	Zhao [126]	64.0	Mun [127]	86.3		
Truc [128]	64.7	Gao [129]	69.6	Yang [130]	69.3	Li [131]	88.1		
Truc [132]	66.1	Zhao [104]	72.6	Waldekar [99]	69.9	Hyder [133]	88.5		
Yang [134]	67.8	Phaye [135]	74.1	Wu [136]	75.4	Song [137]	89.5		
		Heo [138]	77.4	Chen [139]	77.1	Yin [98]	91.0		
log-Mel	58.6	log-Mel	68.0	log-Mel	60.3	log-Mel	80.7	log-Mel	97.9
log-Mel+CQT	64.7	log-Mel+CQT	70.4	log-Mel+CQT	65.8	log-Mel+CQT	89.2	log-Mel+CQT	99.0
log-Mel+CQT+GAM	66.4	log-Mel+CQT+GAM	73.8	log-Mel+CQT+GAM	67.3	log-Mel+CQT+GAM	88.9	log-Mel+CQT+GAM	99.0
log-Mel+CQT+GAM+STFT	66.7	log-Mel+CQT+GAM+STFT	77.3	log-Mel+CQT+GAM+STFT	66.7	log-Mel+CQT+GAM+STFT	88.7	log-Mel+CQT+GAM+STFT	99.1
log-Mel+CQT+GAM+STFT+MFCC	67.5	log-Mel+CQT+GAM+STFT+MFCC	77.8	log-Mel+CQT+GAM+STFT+MFCC	67.0	log-Mel+CQT+GAM+STFT+MFCC	88.2	log-Mel+CQT+GAM+STFT+MFCC	99.1

in Table 9 (Upper part: top-ten DCASE challenges; Middle part: State-of-the-art papers; Low part: Our proposed systems using *Prod* late fusion strategy), we achieve the highest accuracy of 99.1% in LITIS Roune dataset. Our performance in DCASE 2016 is 89.2%, which lies in second position on this challenge table, and is ranked in the top-three of the state-of-the art systems. However, in DCASE 2017, our performance falls outside the top-ten results in this challenge, with the figure of 67.3%. As regards DCASE 2018 Task 1A dataset, the accuracy of 77.8% obtained ranks in top four and exceed all state-of-the art systems. Next, we again show our robustness in terms of dealing with mismatched devices issue in

DCASE 2018 Task 1B. We achieve 67.5% in accuracy rate, outperform systems in DCASE challenge and highly competitive with the top-ranked results in terms of state-of-the-art papers. It should be noted that there are inconsistencies between the reported results in the DCASE 2018 technical reports and those published in DCASE 2018 challenge. The accuracy shown in Tables 9, therefore, are collected from the original sources of technical reports.

3.3 Continuous Wavelet Transform

After the experiments presented in Sections 3.1 and 3.2, it becomes evident that MFCC and log-Mel spectrograms are widely used in audio analysis. However, their dependence on fixed FFT window sizes limits their ability to capture fine temporal and spectral details, which are crucial for effectively distinguishing between normal and abnormal respiratory sounds [28]. Although various spectrogram representations have been used with deep learning models and often outperform hand-crafted features, there is still limited investigation into which spectrogram type is most suitable for lung sound classification.

To address this issue, we adopt multiple wavelet-based scalograms, extending our earlier studies [27, 29, 35, 43, 140]. Wavelet transforms provide multi-resolution analysis by flexibly adjusting temporal windows and capturing energy across a broader frequency range, making them well suited to respiratory acoustics. Based on this idea, Chapter 4 proposes three learning modules—IncF, IncT, and IncFT—to enable the network to learn both global frequency patterns and the temporal distribution of energy within specific frequency bands.

To analyze the respiratory sound signal $y(t)$, a transformation process is applied to convert it into two-dimensional scalograms. This process utilizes the Continuous Wavelet Transform (CWT), employing Amor, Bump, and Morse as the mother wavelet functions [141, 142, 143].

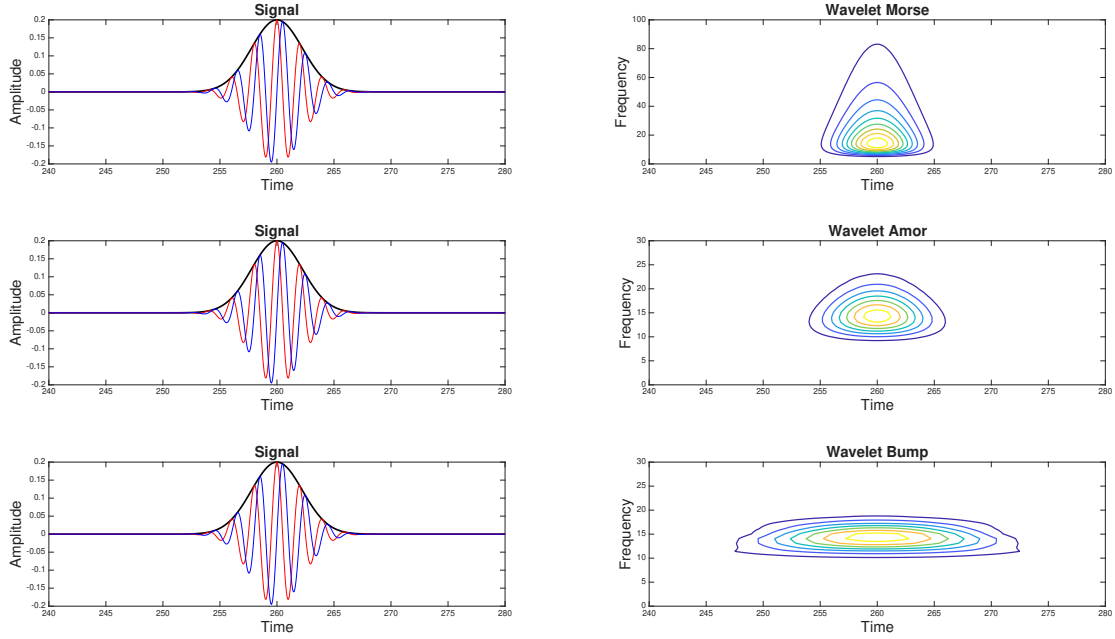


Figure 8: The Frequency and Time resolution in three mother wavelet functions: Morse (Top), Amor (Middle), and Bump (Bottom).

The wavelet transform with the mother wavelet function ψ of a signal $y(t)$ is defined as

$$W_{\psi}^y(a, b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} y(t) \psi\left(\frac{t-b}{a}\right) dt,$$

$$a \in R^+, b \in R,$$

where a is dilatation and b is translation parameter.

3.3.1 Analytic Morlet (Gabor) Wavelet (Amor)

Analytic Morlet (Gabor) Wavelet (Amor): The Analytic Morlet (Gabor) wavelet [142], is a special wavelet for time frequency analysis of acoustic signals. It provides equal variance in time and frequency that is suited to detect suitable time and frequency localized information as presented in the middle part in Figure 8.

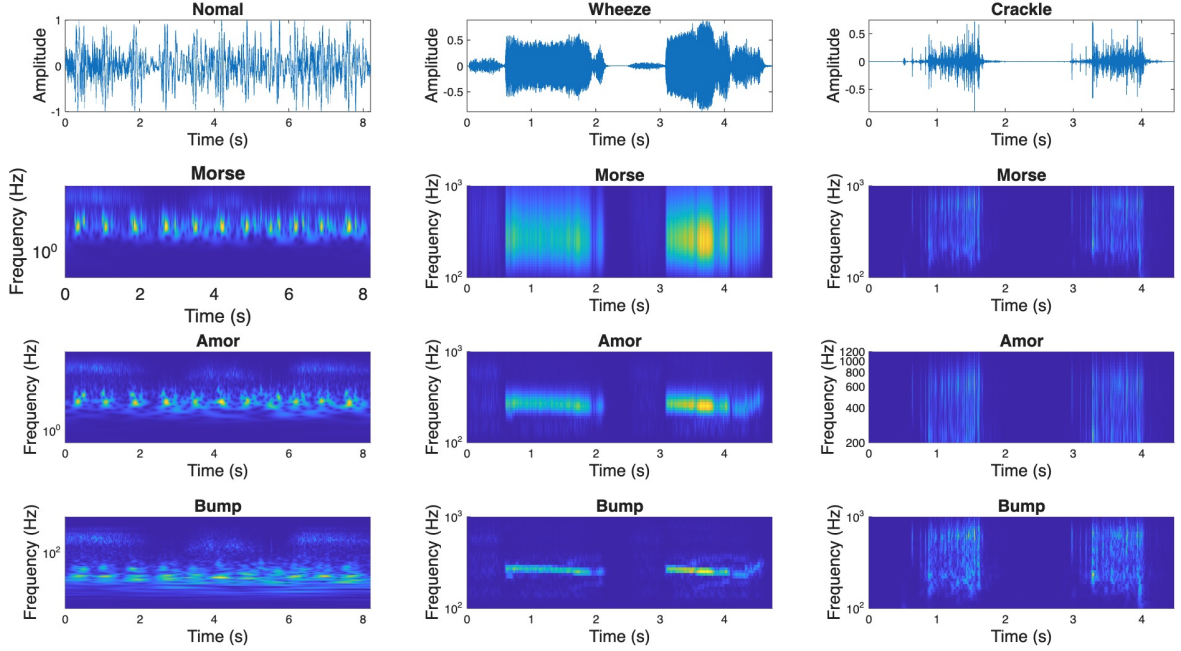


Figure 9: Audio signals and Scalograms of Normal, Wheeze and Crackles.

The Analytic Morlet wavelet ψ is given by

$$\psi(t) = \frac{1}{\sqrt{\pi f_b}} e^{-2\pi i f_c t} e^{-t^2/f_b},$$

where the values of time-decay parameter f_b and center frequency f_c are well defined in [144] to gain excellent time-frequency localization properties.

3.3.2 Generalized Morse Wavelet (Morse)

Generalized Morse Wavelet (Morse): The Morse wavelet [141, 145] is defined as:

$$\psi(t) = \frac{1}{\sqrt{\pi P}} e^{i2\pi f_c t} e^{-t^2/P},$$

where $gamma = 2\pi f_c$, which is the parameter controlling the oscillations of

the complex sinusoid with central frequency of f_c inside the symmetric Gaussian envelope. The time-bandwidth product, P , is proportional to the wavelet duration in time, which determine the wavelet shape and affect the behavior of the transform. In this work, we retain *gamma* parameter at 3 and decrease the time-bandwidth product to 1.5, leading to the more spread out the wavelet is in frequency and narrower the wavelet is in time as illustrated in the top part of Figure 8.

3.3.3 Bump-shaped frequency response wavelet (Bump)

Bump-shaped frequency response wavelet (Bump) [143]: is a band-limited function defined in the frequency domain with parameters μ and σ and window w . In contrast to the Morse, it offers a broader variance in time and a narrower variance in frequency. Valid values for μ are in the range of (3,6) and for σ are in the range of (0.1,1.2). It can be defined as:

$$\hat{\psi}(s\omega) = e^{\left(1 - \frac{1}{1 - \frac{(s\omega - \mu)^2}{\rho^2}}\right)} \cdot I_{\left[\frac{\mu - \sigma}{s}, \frac{\mu + \sigma}{s}\right]}$$

where $I_{[(\mu - \sigma)/s, (\mu + \sigma)/s]}$ is the indicator function for the interval. The value of σ lies in the interval [0.1, 1.2]. For our experiments, we have taken $\mu = 3$ and $\sigma = 1.2$ as the larger values of σ to produce a wavelet with a better time localization as shown in the bottom part in Figure 8.

As a result of these transformations, three types of scalograms are obtained from each respiratory event. All scalograms are scaled to the size of 128×512 (i.e. frequency bands $\times$ the number of time frames) as illustrated in Figure 9.

3.4 Summary on Feature Extraction

Although MFCC and log-Mel spectrograms are widely used techniques, their reliance on fixed window sizes in the FFT limits their ability to capture fine-grained spectral and temporal details [28], which are crucial for distinguishing between normal and abnormal respiratory sounds. While various types of spectrogram have been proposed and used as input to deep learning models and demonstrating superior performance over hand-crafted features, existing studies have not thoroughly explored how to select the most suitable spectrogram type. In particular, there remains a lack of comprehensive analysis on whether a specific type of spectrogram consistently performs well for lung sound classification.

To address this limitation, we propose leveraging multiple scalograms generated via wavelet-based transformations, building on our recent work [27, 29, 35, 43, 140]. This approach provides flexible multi-resolution analysis by adapting temporal window lengths and capturing signal energy over a wider frequency range, which is particularly suited to the acoustic properties of respiratory sounds. Motivated by this, Chapter 4 introduces three dedicated learning modules—IncF, IncT, and IncFT—designed to enable the network to extract both broad frequency-band information and the temporal distribution of energy within specific frequency ranges of the input scalograms.

Chapter 4

Proposed Deep Learning Models

4.1 Deep Learning Framework Applied For Predicting Anomaly of Respiratory Sounds

To continue with the introduction of audio feature extraction in Chapter 3, this Section 4.1 proposes a robust deep learning framework used for classifying anomaly of respiratory cycles. Initially, our framework starts with front-end feature extraction step. This step aims to transform the respiratory input sound into a two-dimensional spectrogram. Next, an ensemble of C-DNN and Autoencoder networks is then applied to classify into four categories of respiratory anomalies.

4.1.1 ICBHI Dataset and Task Definition

The 2017 International Conference on Biomedical Health Informatics (ICBHI) [146] is one of the largest annotated dataset of respiratory sounds published. Specifically, it contains 920 audio recordings collected in several years from 126 subjects in two different European countries. The subjects are identified as being healthy or exhibiting one of the following respiratory diseases or conditions such as: COPD, Bronchiectasis, Asthma, upper and lower respiratory tract infection, Pneumonia,

Bronchiolitis. All recordings account for the duration of 5.5 hours, comprising 6898 respiratory cycles professionally labeled by respiratory experts. Within each audio recording, four different types of respiratory cycle are denoted as *Crackle*, *Wheeze*, *Both* (*Crackle* & *Wheeze*), and *Normal* according to the identified onset (i.e. starting time) and offset (i.e. ending time). Furthermore, these cycles show various duration ranging from 0.2s up to 16.2s and unbalanced (i.e. 1864 cycles of *Crackle*, 886 cycles of *Wheeze*, 506 cycles of *Both*, and 3642 cycles of *Normal*).

Task defined from ICBHI dataset

Table 10: Confusion matrix of anomaly cycle classification.

	Crackle	Wheeze	Both	Normal
Crackle	C_c	W_c	B_c	N_c
Wheeze	C_w	W_w	B_w	N_w
Both	C_b	W_b	B_b	N_b
Normal	C_n	W_n	B_n	N_n
Total	C_t	W_t	B_t	N_t

Given by ICBHI dataset, this thesis evaluates performance of respiratory anomaly classification among four different cycles (*Crackle*, *Wheeze*, *Both*, and *Normal*). In terms of metric used for evaluating, we follow ICBHI challenge, thus report ICBHI scores as mentioned in [146]. In particular, a confusion matrix of respiratory cycle classified is presented in Table 10. Specifically, the letters of C , W , B , and N denote the numbers of cycles of *Crackle*, *Wheeze*, *Both*, and *Normal*, respectively, whereas c , w , b , and n subscripts indicate the inference results. The sums C_t , W_t , B_t and N_t are the total numbers of cycles. Thus, *Sensitivity* (SE), and *Specitivity* (SP) are firstly computed by,

$$Sensitivity = \frac{C_c + W_w + B_b}{C_t + W_t + B_t} \quad (31)$$

$$Specificity = \frac{N_n}{N_t} \quad (32)$$

Table 11: Feature Extraction Parameter Setting.

Factors	Setting
Re-sample	4000 Hz
Cycle duration	10 s
Spectrogram	Gammatonegram [68]
FFT number	1024
Window size	0.2s
Hop size	0.04s
Patch size	128×256
Data augmentation	Random oversampling & Mixup

Next, ICBHI scores comprising average score (AS) and the harmonic score (HS) are computed by,

$$AS = \frac{SE + SP}{2} \quad (33)$$

$$HS = \frac{2.SE.SP}{SE + SP} \quad (34)$$

4.1.2 Front-End Feature Extraction

In particular, we re-sample respiratory cycles to 4000 Hz since frequency bands of abnormal sounds (*Crackle* and *Wheeze*) are primarily located from 60 to 2000 Hz [2, 3, 4]. Consequently, re-sampled respiratory cycles showing different lengths are next duplicated to ensure the same length of 10 seconds. Next, respiratory cycles go through a bandpass filter of 100-2000 Hz to reduce noise. After that, these respiratory sounds are transformed into two-dimensional spectrograms by using Gammatone transformation. To generate Gammatone spectrogram (Gammatone), we firstly compute Short-Time Fourier Transform (STFT). Then, we apply Gammatone filter into STFT spectrogram as described below:

Next, each 10-s spectrogram of one respiratory cycle is thus split into non-overlapped patches of 128×256 , likely an image. To deal with unbalanced data issue, we apply two data augmentation techniques on the image patches of 128×256 .

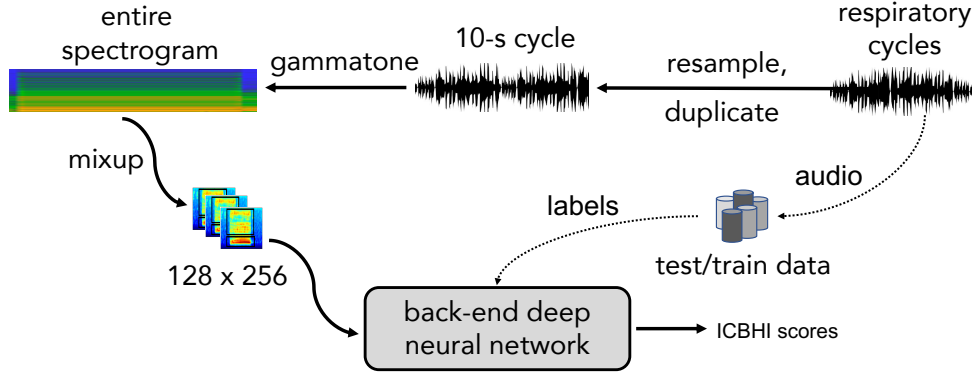


Figure 10: Deep learning based framework proposed.

Firstly, we randomly oversample image patches to make sure that the number of patches per category is equal. Next, the mixup data augmentation [71] is applied to enlarge Fisher's criterion (i.e. the ratio of the between- class distance to the within class variance in the feature space) to increase variation of training data. Let consider two original image patches as $\mathbf{X}_1$, $\mathbf{X}_2$ and expected labels as $\mathbf{y}_1$, $\mathbf{y}_2$, new image patches are generated as below equations:

$$\mathbf{X}_{\text{mp1}} = \mathbf{X}_1\gamma + \mathbf{X}_2(1 - \gamma) \quad (35)$$

$$\mathbf{X}_{\text{mp2}} = \mathbf{X}_1(1 - \gamma) + \mathbf{X}_2\gamma \quad (36)$$

$$\mathbf{y}_{\text{mp1}} = \mathbf{y}_1\gamma + \mathbf{y}_2(1 - \gamma) \quad (37)$$

$$\mathbf{y}_{\text{mp2}} = \mathbf{y}_1(1 - \gamma) + \mathbf{y}_2\gamma \quad (38)$$

where γ is random coefficient from *Beta* distribution, $\mathbf{X}_{\text{mp1}}$, $\mathbf{X}_{\text{mp2}}$ and $\mathbf{y}_{\text{mp1}}$, $\mathbf{y}_{\text{mp2}}$ are new image patches and labels generated, respectively. Eventually, the

mixup patches are fed into a back-end classifier, report the classification accuracy.

4.1.3 Proposed Encoder-Decoder Network Architecture

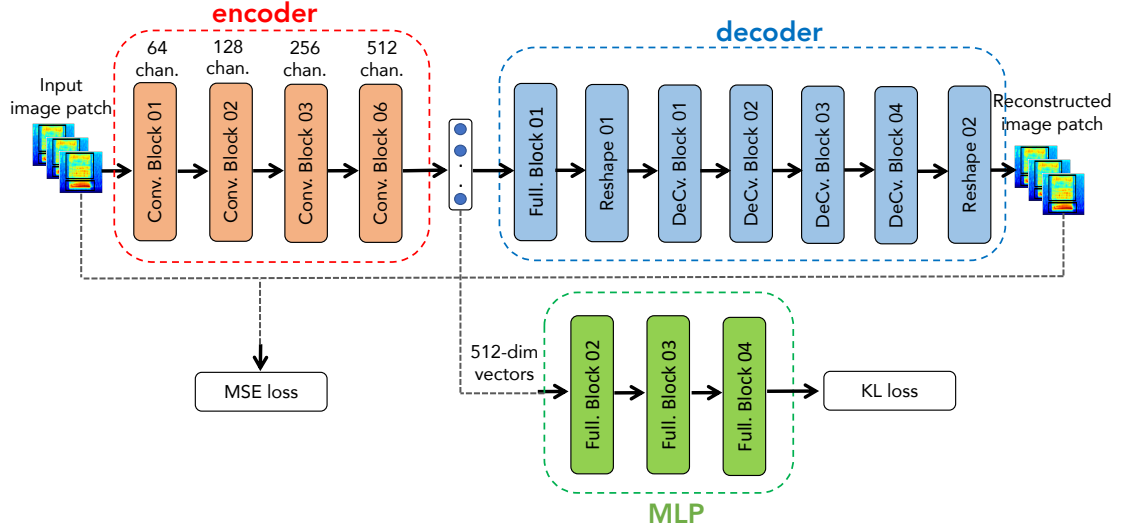


Figure 11: Block-level architecture of Autoencoder network.

Table 12: C-DNN network architecture.

Network architecture	Output
CNN	
Input layer (image patch of 128×256)	
Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (10%)	$62 \times 78 \times 64$
Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (15%)	$31 \times 39 \times 128$
Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (20%)	$16 \times 20 \times 256$
Bn - Cv $[3 \times 3]$ - Relu - Bn - Gmp - Dr (25%)	512
DNN	
Input layer (512-dimensional vectors)	
F1 - Relu - Dr (30%)	1024
F1 - Softmax	4

For back-end classification, we propose an ensemble of C-DNN and autoencoder networks in this thesis to enhance learning on more discriminative features. As regards C-DNN network architecture, it comprises two main parts of CNN and DNN as shown in Table 12, likely Lenet-6 [147]. The CNN as the upper part in

Table 12 performs batch normalization (Bn), convolutional (Cv[kernel size]), rectified linear units (Relu), max pooling (Mp[kernel size]) and global max pooling (Gmp), and dropout (Dr (dropout percentage)) layers. Meanwhile, the DNN as shown in the lower part in Table 12 comprises two fully-connected (Fl), rectified linear units (Relu) and a final Softmax layer for classification.

In terms of autoencoder network architecture proposed, it shows more complicated with two training phases as shown in Figure 11. At the first phase, we present an Encoder-Decoder architecture which is used to extract embedding vectors containing condensed information. Next, the embeddings are fed into a MLP based network architecture for classifying into four categories in the second phase. As shown in Table 13 and the upper part of Figure 11, Encoder part of Encoder-Decoder architecture comprises four Conv. Blocks each which performs layers as same as C-DNN network architecture. The Encoder helps to compress input image patch into condensed vectors, referred to as embeddings. Meanwhile, Decoder firstly use a fully-connected layer to decompress embeddings, thus apply four DeCv. Blocks (i.e. each DeCv. Block comprises a de-convolutional (DeCv[kernel size]) layer and a rectified linear unit layer (Relu)) to re-construct the input image patch). Notably, batch normalization (Bn), max pooling (Mp[kernel size]) and global max pooling (Gmp) and dropout (Dr (dropout percentage)) layers are not applied in Decoder. For MLP-based network architecture as shown in Table 14 and the lower part of Figure 11, it is configured by two fully-connected layers (Fl). The first fully-connected layer follow by a ReLu and a Dr. Meanwhile, a Softmax layer is used after the second fully-connected layer for classification.

4.1.4 Experimental Setting

Given by ICBHI dataset, we follow ICBHI challenge setting, thus divide into Training and Test subsets with the ratio of 60% and 40% respectively as shown in Table 15. Notably, the splitting proposed to ensures that recordings from the

Table 13: Encoder-decoder network architecture.

Block	Network architecture	Output
Encoder		
	Input layer (image patch of 128×256)	
Conv. Block 01	Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (10%)	$64 \times 128 \times 64$
Conv. Block 02	Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (15%)	$32 \times 64 \times 128$
Conv. Block 03	Bn - Cv $[3 \times 3]$ - Relu - Bn - Mp $[2 \times 2]$ - Dr (20%)	$16 \times 32 \times 256$
Conv. Block 04	Bn - Cv $[3 \times 3]$ - Relu - Bn - Gmp - Dr (25%)	512
Decoder		
	Input layer (512-dimensional vectors)	
Full. Block 01	Fl - Relu	32768
Reshape 01	Reshape	$8 \times 16 \times 256$
DeCv. Block 01	DeCv $[3 \times 3]$ - Relu	$16 \times 32 \times 128$
DeCv. Block 02	DeCv $[3 \times 3]$ - Relu	$32 \times 64 \times 64$
DeCv. Block 03	DeCv $[3 \times 3]$ - Relu	$64 \times 128 \times 32$
DeCv. Block 04	DeCv $[3 \times 3]$ - Relu	$128 \times 256 \times 1$
Reshape 02	Reshape	128×256

Table 14: MLP-based network architecture.

Block	Network architecture	Output
	Input layer (512-dimensional vectors)	
Full. Block 02	Fl - Relu - Dr (50%)	1024
Full. Block 03	Fl - Relu - Dr (50%)	1024
Full. Block 04	Fl - Softmax	4

same subject do not appear in both subsets.

In terms for back-end network architectures proposed, we adopt Tensorflow framework and set learning rate to 0.0001, a batch size of 50, epoch number of 100, and Adam method [76] for learning rate optimization. As using mixup data augmentation, the labels are not one-hot format. Therefore, we use Kullback-Leibler (KL) divergence loss [77] in C-DNN and MLP-based networks instead of the standard cross-entropy loss. To train Encoder-Decoder network, we use Mean Squared Error (MSE) loss to compare original image patches (input of Encoder) to reconstructed image patches (output of Decoder) as below,

$$Loss_{MSE} = \frac{1}{2N} \sum_{n=1}^N (\mathbf{X}_n - \hat{\mathbf{X}}_n)^2 \quad (39)$$

Table 15: ICBHI dataset splitting.

	Training Set	Test Set
<i>Wheezes</i>	501	385
<i>Crackles</i>	1215	649
<i>Both</i>	363	143
<i>Normal</i>	2063	1579

where $\mathbf{X}_n$ and $\hat{\mathbf{X}}_n$ are input image patch and reconstructed image patch, respectively.

Late fusion strategy As the C-DNN model work on patch level, the probability of an entire spectrogram is computed by averaging of all patches' probabilities. Let consider $\mathbf{P}_{\mathbf{C-DNN}}^n = (\mathbf{k}_1^n, \mathbf{k}_2^n, \dots, \mathbf{k}_C^n)$, with C being the category number and the n^{th} out of N patches fed into learning model, as the probability of a test sound instance, then the mean classification probability is denoted as $\bar{\mathbf{P}}_{\mathbf{C-DNN}} = (\bar{k}_1, \bar{k}_2, \dots, \bar{k}_C)$ where,

$$\bar{k}_c = \frac{1}{N} \sum_{n=1}^N k_c^n \quad \text{for } 1 \leq n \leq N \quad (40)$$

As the Autoencoder model work on patch level, the probability of an entire spectrogram is computed by averaging of all patches' probabilities. Let consider $\mathbf{P}_{\mathbf{MLP}}^n = (\mathbf{m}_1^n, \mathbf{m}_2^n, \dots, \mathbf{m}_C^n)$, with C being the category number and the n^{th} out of N patches fed into MLP-based network, as the probability of a test sound instance, then the mean classification probability is denoted as $\bar{\mathbf{P}}_{\mathbf{MLP}} = (\bar{m}_1, \bar{m}_2, \dots, \bar{m}_C)$ where,

$$\bar{m}_c = \frac{1}{N} \sum_{n=1}^N m_c^n \quad \text{for } 1 \leq n \leq N \quad (41)$$

To evaluate the ensemble of C-DNN and Autoencoder, we propose three late fusion schemes, namely *Max*, *Mean*, and *Mul* fusions.

The probability of combination with *Max* strategy $\mathbf{p}_{\mathbf{f-max}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C)$ is obtained by,

$$\mathbf{p}_{f-\max} = \max(\bar{\mathbf{p}}_{C-DNN}, \bar{\mathbf{p}}_{MLP}) \quad (42)$$

The probability of combination with *Mean* strategy $\mathbf{p}_{f-\text{mean}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C)$ is obtained by,

$$\mathbf{p}_{f-\text{mean}} = \frac{\bar{\mathbf{p}}_{C-DNN} + \bar{\mathbf{p}}_{MLP}}{2} \quad (43)$$

The probability of combination with *Mul* strategy $\mathbf{p}_{f-\text{mul}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C)$ is obtained by,

$$\mathbf{p}_{f-\text{mul}} = \frac{\bar{\mathbf{p}}_{C-DNN} \cdot \bar{\mathbf{p}}_{MLP}}{2} \quad (44)$$

Eventually, the predicted result is decided by,

$$\hat{y} = \underset{c \in \{1, 2, \dots, C\}}{\operatorname{argmax}} \bar{p}_c. \quad (45)$$

4.1.5 Results and Performance

As details shown in Table 16, it can be seen that Autoencoder is better than C-DNN in terms of SE score, improving by 0.02. By contrast, the SP score of Auto-encoder reduces by 0.001 compared to C-DNN. As a result, both networks achieves same AS score of 0.47. Meanwhile, HS scores present 0.41 and 0.43 for C-DNN and Autoencoder, respectively.

As regards comparison among three late fusion methods, Mean fusion achieves the highest performances with SP score of 0.69, SE score of 0.30, and AS/HS scores of 0.49/0.42. Compared to individual C-DNN or Autoencoder model, although ensemble methods make SE scores reduce a little, they help to improve SP scores significantly.

In comparison with the state-of-the-art systems as shown in Table 17 (note that we only compare to systems which follow splitting ratio of 60/40 defined

Table 16: Performance of C-DNN, Autoencoder, and their fusions (highest scores in **bold**).

Systems	SP	SE	AS/HS Scores
C-DNN	0.63	0.31	0.47/0.41
Autoencoder	0.62	0.33	0.47/0.43
Max fusion	0.67	0.30	0.48/0.42
Mean fusion	0.69	0.30	0.49/0.42
Mul fusion	0.69	0.29	0.49/0.41

Table 17: Compare our systems (the lower part) against state-of-the-art systems with ICBHI challenge splitting (highest scores in **bold**).

Features	Classifiers	SP	SE	AS/HS Scores
MFCC	Decision Tree [148]	0.75	0.12	0.43/0.15
MFCC	HMM [149]	0.38	0.41	0.39/0.23
STFT+Wavelet	SVM [150]	0.78	0.20	0.47/0.24
Gammatonegram	CNN-MoE	0.68	0.26	0.47/0.37
log-Mel	CNN-RNN [33]	0.69	0.30	0.50/ 0.46
Scalogram	CNN-RNN [33]	0.62	0.37	0.50/ 0.46
log-Mel+Scalogram	CNN-RNN [33]	0.81	0.28	0.54/0.42
Gammatonegram	C-DNN	0.63	0.31	0.47/0.41
Gammatonegram	Autoencoder	0.62	0.33	0.47/0.43
Gammatonegram	C-DNN+Autoencoder	0.69	0.30	0.49/0.42

by ICBHI challenge), our system ranks fourth in SP while achieving a top-three position in SE. In terms of only using single spectrogram, our system achieves very competitive AS/HS scores of 0.49/0.42 that ranked second overall just behind the proposed system in [33].

4.2 An Ensemble of Deep Learning Frameworks Applied for Predicting Respiratory Anomalies

In this Section 4.2, we evaluate various deep learning frameworks for detecting respiratory anomalies from input audio recordings. To this end, we firstly transform audio respiratory cycles collected from patients into temporal and spectral representation using the transformation of Wavelet-based scalogram and Log-Mel

spectrogram mentioned in Chapter 3. We then feed these spectrograms and scalograms into back-end deep learning networks for classifying these respiratory cycles into certain categories. Finally, results from high-performed deep learning frameworks are fused to obtain the best score. Our experiments on ICBHI benchmark dataset achieve the highest ICBHI score of 57.3 obtained by a late fusion of inception based and transfer learning based deep learning frameworks, which outperforms the state-of-the-art systems.

To classify four types of respiratory cycles from ICBHI dataset, we firstly propose a high-level architecture of three main deep learning frameworks as shown in Figure 12:

(I) The upper stream in Figure 12 shows how we directly train and evaluate small-footprint inception-based network architectures;

(II) Benchmark and large footprint deep learning network architectures of VGG16, VGG19, MobileNetV1, MobileNetV2, ResNet50, DenseNet201, InceptionV3, Xception are directly trained and evaluated as shown in the middle stream of Figure 12;

(III) The lower stream in Figure 12 shows how we reuse pre-trained models, which were trained with the large-scale AudioSet dataset, to extract embedding features for the training process on multilayer perception (MLP) network architecture. In general, these three proposed deep learning frameworks comprise two main steps of the front-end spectrogram feature extraction and the back-end classification model.

As proposed deep learning frameworks are shown in Figure 12, we firstly duplicate the short-time cycles or cut off the long-time cycles to make all respiratory cycles equal to 10 seconds. For the first two deep learning frameworks (I) and (II), we extract Wavelet-based spectrograms which proves effective in our previous work [151]. As we reuse the setting from [151], we then generate Wavelet

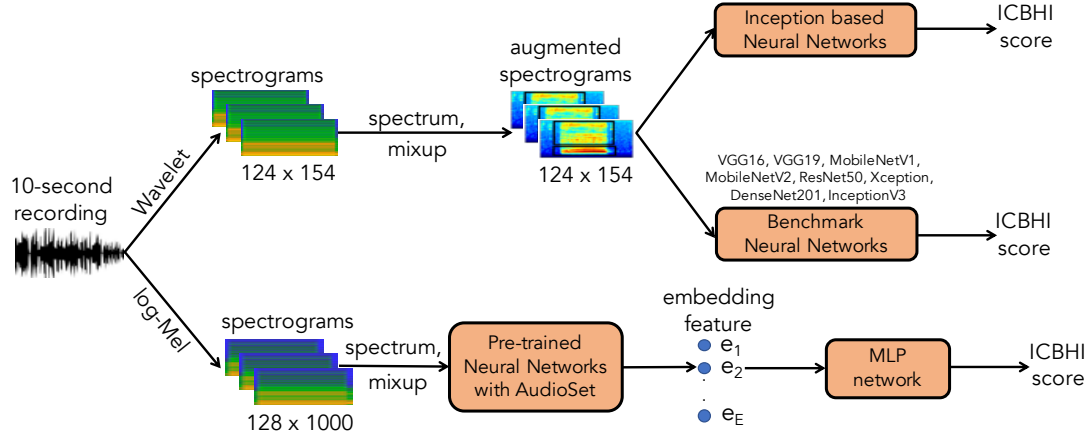


Figure 12: The high-level architecture for three deep learning frameworks proposed.

spectrograms of 124×154 from 10-second respiratory cycles. Meanwhile, we extract log-Mel spectrogram in the deep learning framework (III) as we re-use pre-trained models from [152] receiving log-Mel spectrogram input. By using the same setting from [152], we generate log-Mel spectrograms of 128×1000 for 10-second respiratory cycles. To enforce back-end classifiers, two data augmentation methods of spectrum [153] and mixup [154, 155] are applied on both log-Mel and Wavelet-based spectrograms before feeding into back-end deep learning models for classification.

4.2.1 The Low-footprint Inception Based Network architectures

(I) *The low-footprint inception based network architectures:* As the potential results were achieved from our previous work [151], we further evaluate different inception based network architectures in this thesis to evaluate the potential of light weight model, especially for embedded and edge devices in CRAA. In particular, two high-level architectures with single or double inception layers as shown in Table 18 are used in this thesis. These architectures perform different layers:

Table 18: The general inception-based network architectures.

Single Inception Layer	Double Inception Layers
BN	
Inc(ch1) - ReLU	Inc(ch1) - ReLU - Inc(ch1) - ReLU
BN - MP - Dr(10%) - BN	
Inc(ch2) - ReLU	Inc(ch2) - ReLU - Inc(ch2) - ReLU
BN - MP - Dr(15%) - BN	
Inc(ch3) - ReLU	Inc(ch3) - ReLU - Inc(ch3) - ReLU
BN - MP - Dr(20%) - BN	
Inc(ch4) - ReLU	Inc(ch4) - ReLU - Inc(ch4) - ReLU
BN - GMP - Dr(25%)	
FC(fc1) - ReLU - Dr(30%)	
FC(fc2) - ReLU - Dr(30%)	
FC(4) - Softmax	

inception layer (Inc(output channel number)) as shown in Figure 13, batch normalization (BN) [156], rectified linear units (ReLU) [157], max pooling (MP), global max pooling (GMP), dropout [158] (Dr(percentage)), fully connected (FC(output node number)) and Sigmoid layers. By using the two architectures and setting different parameters to channel numbers of inception layers and output node numbers of fully connected layers, we then create six inception based deep neural networks as shown in Table 19, referred to as Inc-01, Inc-02, Inc-03, Inc-04, Inc-05, and Inc-06, respectively.

4.2.2 The Benchmark and Complex Neural Network Architectures

We further evaluate different benchmark neural network architectures of VGG16, VGG19, MobileNetV1, MobileNetV2, ResNet50, DenseNet201, InceptionV3, and

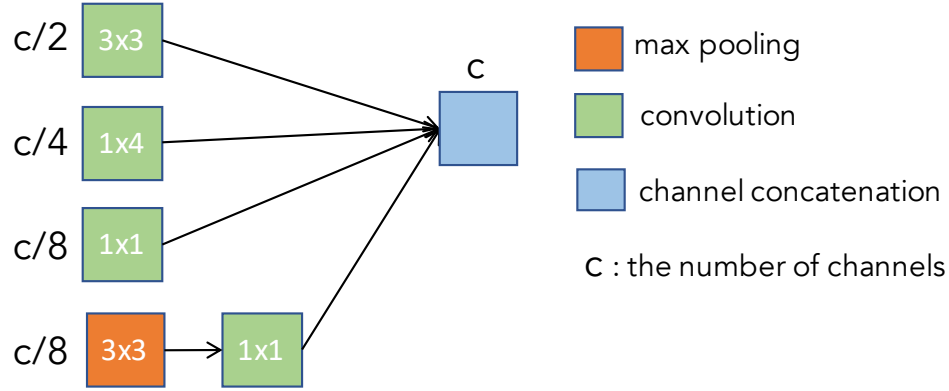


Figure 13: The single inception layer architecture.

Table 19: Settings for inception-based network architectures.

Hyper-parameter	Inc-01	Inc-02	Inc-03	Inc-04	Inc-05	Inc-06
Inception Layer Type	Single	Double	Single	Double	Single	Double
ch1	32	32	64	64	128	128
ch2	64	64	128	128	256	256
ch3	128	128	256	256	512	512
ch4	256	256	512	512	1024	1024
fc1	512	512	1024	1024	2048	2048
fc2	512	512	1024	1024	2048	2048

Xception, which are available in Keras library [159] and popularly applied in different research domains. In comparison with the inception based network architectures used in the framework (I), these benchmark neural networks show large footprint with complex architecture and trunks of convolutional layers.

4.2.3 The Transfer Learning Based Network Architectures

As transfer learning techniques have proven effective for downstream tasks with a limitation of training data and smaller categories classified [152], we leverage pre-trained networks which were trained with the large-scale AudioSet dataset [152]: LeeNet24, DaiNet19, VGG14, MobileNetV1, MobileNetV2, Res1DNet30, ResNet38,

Wavegram-CNN. We then modify these networks to match the downstream task of classifying four respiratory cycles in ICBHI dataset. This motivation is based on to evaluate whether this transfer learning is especially beneficial for downstream ICBHI dataset, which is relatively small and imbalanced. In particular, we retain trainable parameters from the first layer to the global pooling layer of the pre-trained networks that originally trained on the large-scale AudioSet. We then replace layers after the global pooling layer by new fully connected layers to create a new network (i.e. Trainable parameters in new fully connected layers are initialized with mean and variance set to 0 and 0.1, respectively). In the other words, we use a multilayer perception (MLP) as shown in Table 21 (i.e. The multilayer perception (MLP) is configured by FC, ReLU, Dr, and Softmax layers) to train embedding features extracted from the pre-trained models (i.e. The embedding features are the feature map of the final global pooling layer in the pre-trained networks).

4.2.4 Early, Middle, and Late Fusion of Inception Based and Transfer Learning Based Frameworks

As using spectrum [153] and mixup [154, 155] data augmentation methods, labels are not one-hot encoding format. Therefore, we use Kullback–Leibler divergence (KL) loss. While we construct deep learning networks proposed in frameworks (I) and (II) with Tensorflow, we use Pytorch for extracting embedding features and training MLP in the framework (III) as the pre-trained networks were built in Pytorch environment. We set epoch number=100 and using Adam method [160] for optimization.

As the deep learning frameworks basing on Inc-03 and transfer learning with the pre-trained VGG14 achieve the best scores, we then evaluate whether a fusion of results from these frameworks can help to further improve the task performance. In particular, we propose three fusion strategies:

In the first and second fusion strategies referred to as the early and middle fusion, we concatenate the embedding feature extracted from the pre-trained VGG14 (e.g. the feature map of the global pooling of the pre-trained VGG14) with an embedding feature extracted from Inc-03 to generate a new combined feature. We then train the new combined feature with a MLP network architecture as shown in Table 21. While the feature map of the max global pooling (MGP) of Inc-03 is considered as the embedding feature in the first fusion strategy, the feature map of the second fully connected layer of Inc-03 (e.g. FC(fc2)) is used for the second fusion strategy.

In the third fusion strategy referred to as the late fusion, we use PROD fusion of the predicted probabilities obtained from these inception-based and transfer learning based frameworks. The PROD fusion result $\mathbf{p}_{f-\text{prod}} = (\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C)$ is obtained by:

$$\bar{p}_c = \frac{1}{S} \prod_{s=1}^S \bar{p}_{sc} \quad \text{for } 1 \leq s \leq S, \quad (46)$$

where $\bar{\mathbf{p}}_s = (\bar{p}_{s1}, \bar{p}_{s2}, \dots, \bar{p}_{sC})$ is the predicted probability of a single framework, C is the category number and the s^{th} out of S individual frameworks evaluated. The predicted label $\hat{y}$ is determined by:

$$\hat{y} = \text{argmax}(\bar{p}_1, \bar{p}_2, \dots, \bar{p}_C) \quad (47)$$

As Table 22 shows, all three fusion strategies help to enhance the performance, report a further ICBHI score improvement of 0.3, 1.1, 2.2 from early, middle, late fusion strategies, respectively. It proves that embedding features extracted from the inception based Inc-03 framework and the transfer learning framework with the pre-trained VGG14 contain distinct features of respiratory cycles.

4.2.5 Results and Performance

Performance comparison among deep learning frameworks proposed

As the experimental results are shown in Table 20, it can be seen that generally the low-footprint inception based frameworks and the transfer learning based frameworks are competitive and outperform the benchmark frameworks. Table 20 records the best ICBHI score of 55.1% from the Inc-03 framework and the transfer learning framework with the pre-trained VGG14 (i.e. The best performance obtained from the pre-trained VGG14 makes sense as this network outperforms the other network architectures for classifying sound events in AudioSet dataset). Notably, while we use the same network architecture of MobileNetV1 and MobinetV2, transfer learning frameworks significantly outperform the benchmark frameworks. As a result, we can conclude that (1) applying the transfer learning technique on the downstream task of classifying respiratory cycles is effective; and (2) low-footprint inception based networks focusing on minimal variation of time and frequency are effective for respiratory sounds rather than benchmark and large network architectures.

Performance comparison to the state of the art To compare with the state of the art, we only select published systems which follow the recommended official setting of ICBHI dataset [161] with the train/set ratio of 60/40 and none of overlapping patient subject on both subsets. As experimental results are shown in Table 23, our system with a late fusion of inception-based and transfer learning frameworks outperform the state of the art, records the best score of 57.3%.

4.3 An Inception-Residual-Based Architecture with Multi-Objective Loss for Detecting Respiratory Anomalies

This section 4.3 presents a deep learning system applied for detecting anomalies from respiratory sound recordings. Initially, our system begins with audio feature extraction using Gammatone and Continuous Wavelet transformation as presented in Chapter 3. This step aims to transform the respiratory sound input into a two-dimensional spectrogram where both spectral and temporal features are presented. Then, our proposed system integrates Inception-residual-based backbone models combined with multi-head attention and multi-objective loss to classify respiratory anomalies. Instead of applying a simple concatenation approach by combining results from various spectrograms, we propose a Linear combination, which has the ability to regulate equally the contribution of each individual spectrogram throughout the training process. To evaluate the performance, we conducted experiments over the benchmark dataset of SPRSound (The Open-Source SJTU Paediatric Respiratory Sound) proposed by the IEEE BioCAS 2022 challenge

4.3.1 SPR Dataset and Task Defined

In this work, we evaluate our systems on the benchmark 2022 SPRSound: Open-Source SJTU Paediatric Respiratory Sound database, which was collected in the Shanghai Children’s Medical Center (SCMC), China [60]. The dataset consists of 2683 audio recordings collected from 292 patients (whose age is from 1 month to 18 years old). After being carefully inspected by experts, the quality of each recording was labeled as *Poor Quality (PQ)*, *Normal (N)*, *Continuous Adventitious Sound (CAS)*, *Discontinuous Adventitious Sound (DAS)*, and *CAS and DAS (CD)*

with the recording numbers of 187, 1785, 233, 347, and 131, respectively. For each audio event in a recording, the onset (i.e. starting time) and offset (i.e. ending time) are professionally labeled by respiratory experts. There are total of 6887 audio events of *Normal* (N), 53 audio events of *Rhonchi* (Rho), 865 audio events of *Wheeze* (W), 17 audio events of *Stridor* (Str), 66 audio events of *Coarse Crackle* (CC), 1167 audio events of *Fine Crackle* (FC) and 34 audio events of *Wheeze and Crackle* (B).

Overall, there is an issue of imbalance between audio events as well as entire recordings. Additionally, these audio recordings and events show various duration ranging from 0.304s-15.36s and 0.126s to 7.152s, respectively, which makes the 2022 SPRSound dataset more challenging.

Tasks Definition

Given the SPRSound dataset, the IEEE BioCAS 2022 challenge proposed two main tasks.

Task 1, which aims to focus on the sound event level, is divided into 2 sub-tasks: Task 1-1 and Task 1-2. While Task 1-1 is to classify the respiratory sound events as Normal and Adventitious, Task 1-2 aims to classify these events into N , Rho , W , Str , CC , FC , or B .

Task 2 focuses on the entire recording, which is also separated into Task 2-1 and Task 2-2. In particular, Task 2-1 accounts for classifying the respiratory recordings as Normal, Adventitious, and Poor Quality. Meanwhile, Task 2-2 is a multi-class classification, where the respiratory recordings are classified into N , CAS , DAS , CD , or PQ . To adhere to the evaluation metrics in the IEEE BioCAS 2022 challenge, every task and its sub-tasks in this thesis are evaluated by *sensitivity* (SE), *specificity* (SP), *average score* (AS), *harmonic score* (HS), and the average of AS and HS (Score).

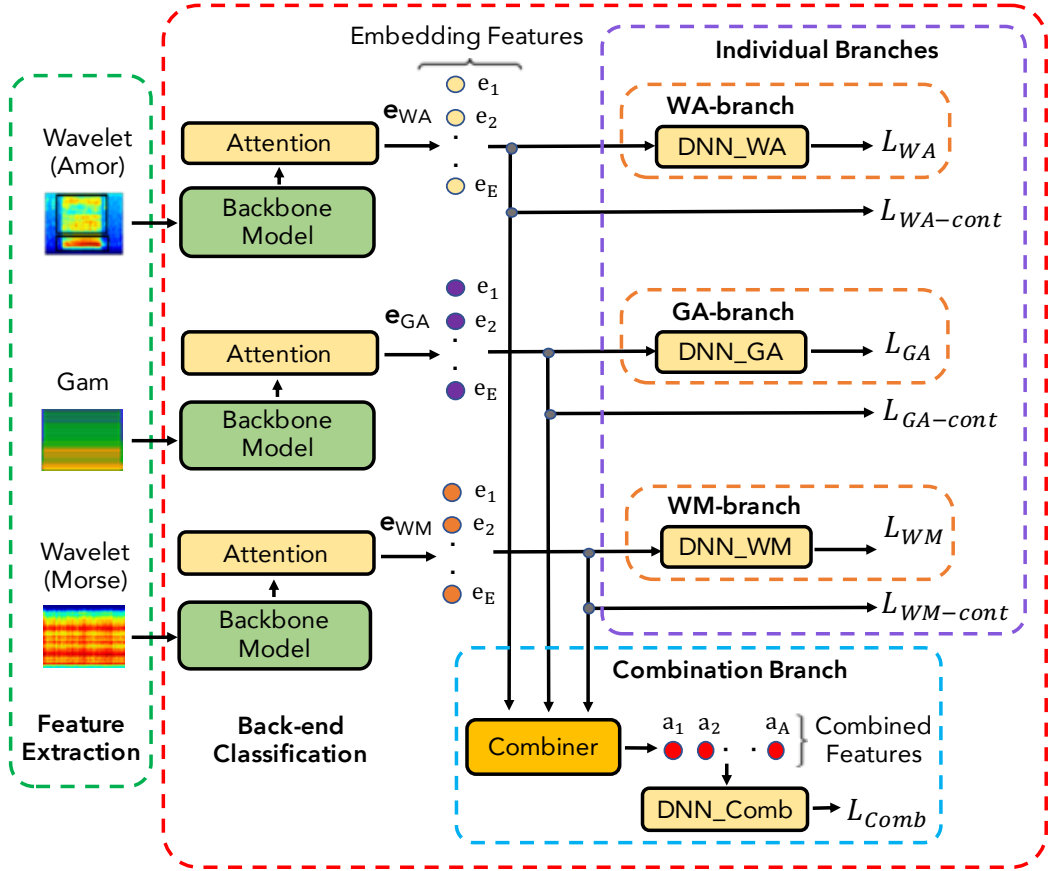


Figure 14: The high-level architecture of our proposed system for detecting anomalies in respiratory audio recording.

Overall, our proposed system for detecting anomalies in respiratory sounds as shown in Figure 14 consists of two main stages: low-level spectrogram feature extraction and back-end classification.

4.3.2 Front-End Feature Extraction

At the sound event level, the respiratory events evaluated in Task 1 are re-sampled to 4 kHz as all abnormal sounds have their frequency bands located around 60-2000 Hz [60]. Next, re-sampled respiratory events presenting different recording lengths are then duplicated to obtain a uniform duration of 10 seconds. A band-pass filter of 60-2000 Hz is then applied to suppress background noise in every respiratory event. After that, these respiratory events are transformed into a

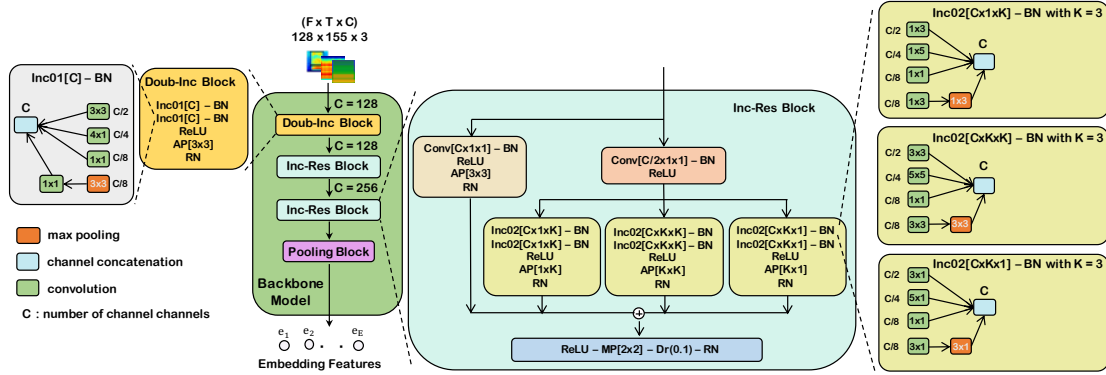


Figure 15: The detailed architecture of the Backbone Model used in the proposed system.

two-dimensional spectrogram by using Gammatone [167] and Continuous Wavelet transformations. For the Continuous Wavelet transformations, we use Amor and Morse as the mother wavelet functions. As a result, we obtain three spectrograms from one sound event. Finally, all spectrograms are scaled into the size of 128×155 (i.e. frequency bands $\times$ the number of time frames).

At the recording level for evaluation in Task 2, the entire audio recordings are duplicated to make sure that all audio recordings have a consistent duration of 15.36 seconds in accord with the longest recording. Similar to Task 1, these audio recordings are then transformed into Gammatone spectrogram and Wavelet-based spectrograms. Finally, entire spectrograms are scaled to the sizes of 128×512 (i.e. the longer duration of the entire recording results in more time frames compared to the event level).

After generating spectrograms, we apply three data augmentation methods on the spectrograms on both event and entire recording levels to deal with the imbalanced data issue:

First, we randomly oversample spectrograms to make sure that the number of spectrograms per class is equal in each batch size.

Second, the mixup data augmentation [168] is applied to increase the variation

of the training data and to enlarge Fisher’s criterion (i.e. the ratio of the between-class distance to the within-class variance in the feature space).

Third, augmented spectrograms in every batch are randomly cropped with a reduction of 10 bins on both time and frequency dimensions to motivate learning toward the partial losses of information at each dimension [169].

4.3.3 Inception-residual-based Backbone Models

The back-end classifier comprises four main branches. The first three branches comprise **WA-branch**, **GA-branch**, and **WM-branch**, which aim to focus on exploring individual spectrogram inputs as shown in the upper part of Figure 14. We denote the spectrogram generated by Gammatone transformation as GA, whereas we mark WA and WM as obtained spectrograms from Amor and Morse on the mother wavelet functions in CWT, respectively. Meanwhile, the final branch, referred to as **Combination branch**, collects embedding features extracted from the first three individual branches (i.e. the outputs of Attention blocks) to derive a Combined feature as shown in the lower part of Figure 14.

Individual branches of **WA-branch**, **GA-branch**, and **WM-branch** comprises a Backbone Model followed by an Attention block and a DNN-based block (i.e. DNN_WA, DNN_GA, or DNN_WM). In particular, the detailed architecture of the Backbone Model is illustrated in Figure 15. The proposed Backbone Model based on Inception-residual-based network architecture comprises four blocks: one Doub-Inc Block, two Inc-Res Blocks, and one Pooling Block. The Doub-Inc Block performs two Inc01 layers followed by Batch Normalization (BN), Rectified Linear Unit (ReLU), Average Pooling (AP[Kernel Size]), Dropout (Dr(Drop Ratio), and Residual Normalization (RN($\lambda = 0.4$)) inspired from [170]). The Inc01 layer is a variant of the naive inception layer [171] with fixed kernel sizes of $[3 \times 3]$, $[1 \times 1]$, and $[4 \times 1]$. Regarding two Inc-Res Blocks, they share the same architecture, but channel numbers increase from 128 to 256 to provide a deeper representation

of the channel dimension as inspired by our prior work in [172]. The detailed structure for every Inc-Res Block is described in the middle part of Figure 15, which features layers of Inc02[Channel $\times$ Kernel Size], Conv, BN, Dr, ReLU, AP, Max Pooling (MP[Kernel Size]), and RN($\lambda = 0.4$). Notably, three different Inc02[Channel $\times$ Kernel Size] layers, which present different kernel sizes defined as $[K \times 1]$, $[K \times K]$, and $[1 \times K]$, are used in Inc-Res Block. The value of K is changed as detailed on the right-hand side of Figure 15. Following each Inc02 layer, an AP layer is applied before adding sub-branch results together. Furthermore, we propose to apply RN to form the Inc-Res Block as RN can be used as instance normalization with a shortcut path to eliminate the influence of unnecessary responses from some frequency bands during training (i.e. these influences caused by noises from environments like coughing, motion artifacts, and intestinal and physiological sounds when respiratory sounds are recorded). This idea strengthens the backbone to learn effectively not only the widespread frequency bands but also the distribution of energy in certain frequency bands from different types of anomalies in lung sound. The Pooling Block makes use of global pooling layers to extract three features from the second Inc-Res Block: (1) global average pooling across the channel dimension, (2) global max pooling across the temporal dimension, and (3) global average pooling across frequency dimensions.

As the work in [173] indicates that each attention head learns a different set of weight matrices and when joining many self-attention heads together, it forms a multi-head self-attention layer which helps to learn an input feature map better. Therefore, in this work, we propose that the output of the Backbone Model (i.e. The output of the Pooling Block in Figure 15) is then presented to the Attention block as shown in Figure 14. At each Attention block, we apply three multi-head attention layers [173] on three dimensions of frequency, time, and channel to fully exploit the embeddings extracted from different aspects of the spectrograms

(i.e. specific embeddings extracted from different spectrograms). Each multi-head attention layer is configured to have 16 as the number of heads and 32 as the key dimension. The output of each multi-head attention layer is a one-dimensional embedding feature (i.e. the embedding feature is described as a vector $\mathbf{e}[e_1, e_2, \dots, e_E]$ in Figure 14).

The embedding features from Attention Blocks are finally fed into individual DNN_WA, DNN_GA, or DNN_WM blocks for classification. The blocks of DNN_WA, DNN_GA, and DNN_WM share the same network architecture which comprises two dense layers. The first dense layer comprises a fully connected layer (FC[C = E]) layer followed by BN, ReLU, and Dr, where E is the dimensional length of the embedding features. Meanwhile, the second dense layer comprises a fully connected layer (FC[C = T]) followed by a Softmax, where T is defined according to the number of target classes.

Combination branch architecture: This final branch as shown in the lower part of Figure 14 comprises two main blocks: Combiner and DNN_Comb. The Combiner block gathers embedding features $\mathbf{e}[e_1, e_2, \dots, e_E]$ outputted from Attention blocks to obtain a Combined feature $\mathbf{a}[a_1, a_2, \dots, a_E]$. As the use of multiple spectrograms has shown improvement as mentioned in Section 4.3.2, we propose this Combiner block. We assume that each spectrogram contains distinct features, and then the Combined feature from the Combiner block are potentially better representing features from all the spectrograms for detecting anomalies in respiratory sounds. In this work, we propose two combination methods. The first method is simple concatenation. In the second method, we assume that the feature embeddings $\mathbf{e}[e_1, e_2, \dots, e_E]$ from each spectrogram have a linear relationship across their dimensions. We then derive a data-driven combination method called Linear combination. This Linear combination is proposed to combine the three high-level features with trainable weight and bias:

$$\begin{aligned} \mathbf{a}_{Linear-combined} = ReLU(\mathbf{e}_{WA}\mathbf{w}_{WA} + \\ \mathbf{e}_{GA}\mathbf{w}_{GA} + \mathbf{e}_{WM}\mathbf{w}_{WM} + \mathbf{w}_{bias}), \end{aligned} \quad (48)$$

where $\mathbf{w}_{WA/GA/WM/bias}[w_1, w_2, \dots, w_E]$ are the trained parameters. The Combined feature $\mathbf{a}$ is finally fed into the DNN_Comb block for classification. The DNN_Comb block presents the same architecture as DNN_GA, DNN_WA, or DNN_WM blocks.

Training loss functions: we propose different loss functions to train the proposed system. In particular, we use two loss functions of KL-loss and contrastive loss [174] for each Spec-Ind branch to learn individual spectrogram input. First, as using mixup data augmentation, the labels are not one-hot format. Therefore, we use Kullback-Leibler (KL) divergence loss in the proposed networks. We set the learning rate to 0.0001 and Adam method [160] is applied for optimization. While KL-loss functions of L_{WA} , L_{GA} , L_{WM} help to classify target classes in each individual spectrogram branch, KL-loss of L_{Comb} is used in the Combination branch.

Second, we also apply contrastive loss in our training process. We manage to have a pair of spectrograms (S_i, S_j) and a label Y . The label is equal to 1 if the spectrograms are in the same class and 0 otherwise. To extract embedding features of each spectrogram, we use the Backbone model f that encodes the spectrogram S_i and S_j into an embedding space where $\mathbf{e}_i = f(S_i)$ and $\mathbf{e}_j = f(S_j)$. The contrastive loss is defined as:

$$L_{Cont} = Y * d^2 + (1 - Y) \cdot \max(\text{margin} - d, 0)^2, \quad (49)$$

where $d = \|\mathbf{e}_i - \mathbf{e}_j\|_2$ is the Euclidean distance between embeddings and margin is set to 1. In every spectrogram branch, the contrastive loss functions of

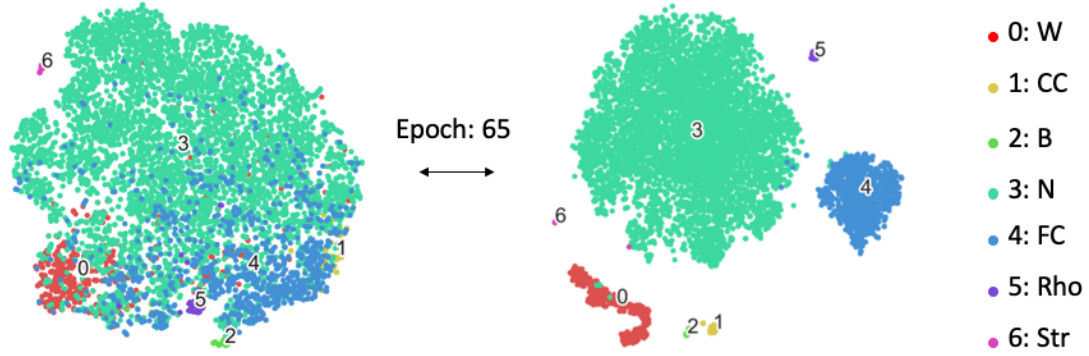


Figure 16: The efficiency of applying contrastive loss in the training process on Task 1-2.

$L_{WA-Cont}$, $L_{GA-Cont}$, and $L_{WM-Cont}$ are applied to embedding features, which helps to maximize the Euclidean distance between embeddings from different classes while minimizing the Euclidean distance between embeddings from the similar classes.

4.3.4 Multi-Objective Loss

Eventually, the multi-objective loss is computed as:

$$L_{total} = \alpha(L_{WA} + L_{GA} + L_{WM}) + \beta(L_{Comb}) + \gamma(L_{WA-Cont} + L_{GA-Cont} + L_{WM-Cont}), \quad (50)$$

where α , β , and γ are loss weight ratios that are used to manage the contribution of every single objective loss.

4.3.5 Results and Performance

As we use multiple techniques, including multi-branch structures, multi-head attention, features combinations, and multi-objective loss function to construct our proposed system as shown in Figure 14, we then evaluate the individual role of certain techniques by defining four system variants:

Individual branch: This system only uses one backbone model and one of three branches: WA-branch, GA-branch, or WM-branch. This system is used to evaluate the contribution of every individual spectrogram.

System I: This system uses all four branches, the concatenation method is used in the Combiner block, but does not use multi-head attention and contrastive losses. The values of loss weight ratios of α , β and γ are empirically set to 1/3, 1, and 0, respectively.

System II: This system reuses the settings of System I but adds Attention blocks. The values of loss weight ratios of α , β and γ are empirically set to 1/3, 1, and 0, respectively.

System III: This system uses all techniques mentioned in this thesis (i.e. employ Attention block, Linear combination method in the Combiner block, both KL-loss and contrastive loss). The values of loss weight ratios of α , β and γ are empirically set to 1/3, 1, and 1, respectively.

As Table 24 shows, in each individual spectrogram branch, tasks at the event level (i.e. Task 1-1, Task 1-2) have a better performance compared to tasks in the recording level (i.e. Task 2-1, Task 2-2). This can be explained as Task 1-1 and 1-2 only focus on certain events, while Task 2-1 and 2-2 focus on the entire recording including different types of events, and poor quality recordings are treated as one of the categories in Task 2-1 and Task 2-2. Among individual spectrograms, WM has the best performance with HS of 80.9%, 68.2%, 63.4%, and 38.7% in Task 1-1, Task 1-2, Task 2-1, and Task 2-2, respectively.

In Table 25, System I with multiple spectrogram inputs indeed improves the performance in most tasks compared to that in individual WA-branch and GA-branch. However, Task 2-2 in System I witnesses an overfitting performance of 91.7% in SP. Notably, although System I uses multiple spectrograms with simple concatenation, there is even a decrease in performance in Task 1-1 with lower AS/HS of 81.2%/80.6% compared to 81.5%/80.9% in the best single spectrogram

WM-branch.

When the multi-head attention technique is applied as mentioned in System II, all performance criteria of SE/SP and AS/HS are improved compared with System I and single spectrogram branches. The issue of overfitting on SP in Task 2.2 of System I is also resolved. For instance, the performance of HS in Task 2-1 and Task 2-2 witnesses an improvement of 6.5% and 7.1%, respectively, compared to that in System I.

Regarding System III, applying both Multi-head attention and a Linear combination of embedding features helps to further improve the performance and proves that System III achieves the best performance compared to the other evaluating systems. As shown in Table 25, the highest AS/HS of 84.9%/84.9% are observed in Task 1-1. On Task 1-2, this system also surpasses other systems in terms of AS and HS, with the results being 78.1% and 76.7%. Notably, the superiority of this system is evident as HS in Task 2-2 witnesses a significant improvement of 18.2% and 11.1% compared with that in System I and System II, respectively. This again demonstrates that multi-head attention and linear combination play a significant role in increasing performance. In addition, the implementation of contrastive loss in System III helps to significantly reduce the training time and comprehensively discriminate the distribution of embedding features among target classes. For instance, Figure 16 shows that System III with contrastive loss converges faster at the same epoch of 65 compared to the same system without using the contrastive loss in Task 1-2. (On the left: t-SNE map of embeddings extracted from System III without contrastive loss. On the right: t-SNE map of embeddings extracted from System III with contrastive loss).

Compared to the Challenge baseline and top-3 systems submitted for the IEEE BioCAS 2022 challenge as shown in Table 26 (i.e. these systems just reported the Scores), we gain Scores of 84.9% in Task 1-1 and 77.4% in Task 1-2 while the highest Scores are 89.0% and 82.0%, respectively. Significantly, our proposed

System III has demonstrated more efficiency than others as it outperforms the state-of-the-art in Task 2-1 and Task 2-2, achieving the highest Score of 74.5% and 53.9%. However, the low performances on Task 2-2 compared to Task 1-2 in Table 26 present a challenge for the task of multi-class classification on the entire recording level.

4.4 A Deep Learning Architecture with Spatio-Temporal Focusing for Detecting Respiratory Anomalies

This section 4.4 presents a deep learning system applied for detecting anomalies from respiratory sound recordings. Our system initially performs audio feature extraction using Continuous Wavelet transformation. This transformation converts the respiratory sound input into a two-dimensional spectrogram where both spectral and temporal features are presented. Then, our proposed deep learning architecture inspired by the Inception-residual-based backbone performs the spatio-temporal-focusing and multi-head attention mechanism to classify respiratory anomalies. In this work, we evaluate our proposed models on the benchmark SPRSound database proposed by the IEEE BioCAS 2023 challenge. Overall, our proposed system for detecting anomalies in respiratory sounds as shown in Figure 17 consists of three main steps: low-level spectrogram feature extraction, data augmentation, and back-end classification.

4.4.1 IEEE BioCAS 2023 grand challenge and Tasks Defined

In this work, we utilize the 2022 SPRSound: Open-Source SJTU Paediatric Respiratory Sound database, which was collected at the Shanghai Children’s Medical

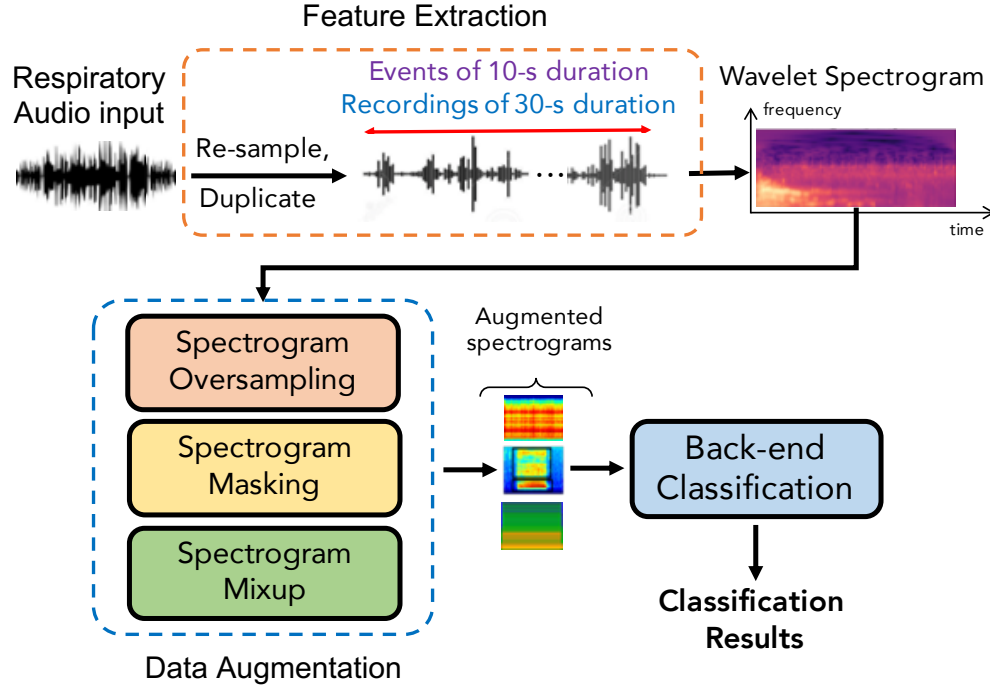


Figure 17: The high-level architecture of our proposed system for detecting anomalies in respiratory audio inputs.

Center, China [60]. The database comprises 2,683 audio recordings obtained from 292 patients aged between 1 month and 18 years old. These audio recordings were recorded using a Yunting model II Stethoscope at a sampling rate of 8 kHz with 16-bit precision. After a careful examination by experts, each recording was labeled as Poor Quality (PQ), Normal (N), Continuous Adventitious Sound (CAS), Discontinuous Adventitious Sound (DAS), or CAS and DAS (CD). In addition, respiratory experts also annotated the onset (starting time) and offset (ending time) of every audio event across the recordings. As a result, this database consists of audio events classified as Normal (N), Rhonchi (Rho), Wheeze (W), Stridor (Str), Wheeze and Crackle (B), Coarse Crackle (CC), and Fine Crackle (FC). Furthermore, these audio recordings and events show various duration ranging from 0.3 seconds to 15.4 seconds and 0.1 seconds to 7.2 seconds, respectively. Moreover, the imbalance in the distribution among classes in both recording and event levels makes the classification more challenging.

There are two levels (i.e. event and recording) considered in classification tasks in the SPRSound database, referred to as Task 1 and Task 2. **Task 1** comprises two sub-tasks of Task 1-1 and Task 1-2. While Task 1-1 is to classify the respiratory sound events as Normal and Adventitious, Task 1-2 involves classifying these events into *N*, *Rho*, *W*, *Str*, *CC*, *FC*, or *B*. **Task 2** focuses on the entire recording, which is also separated into Task 2-1 and Task 2-2. In particular, Task 2-1 aims to classify the respiratory recordings as Normal, Adventitious, and Poor Quality. Meanwhile, Task 2-2 is a multi-class classification, where the respiratory recordings are classified into *N*, *CAS*, *DAS*, *CD*, or *PQ*. We adhere to the evaluation metrics as mentioned in [60] where every task and its sub-tasks are evaluated by SE, SP, AS, HS, and the average of AS and HS. As the SPRSound database is proposed in the IEEE BioCAS 2023 challenge, we follow the challenge instructions to separate the database into Training and Validation sets. After evaluating our proposed models on the Validation set, we submit the models to the challenge for evaluation on a blind Test set.

4.4.2 Data Augmentation

To address the issue of imbalanced data, we propose three data augmentation techniques for both the event and entire recording levels after spectrograms are generated from the feature extraction step. First, we randomly oversample the spectrograms to ensure there is an equal number of spectrograms per class in each batch size. This helps to balance the representation of different classes in the training data. Second, spectrograms within each batch are then randomly cropped with a reduction of 10 bins in both the time and frequency dimensions. This encourages the learning process to focus on the partial loss of information at each dimension [169]. Third, we apply the mixup data augmentation technique [168] to increase the variation in the training data. This technique combines pairs of spectrograms from different classes, creating new synthetic samples. The goal is

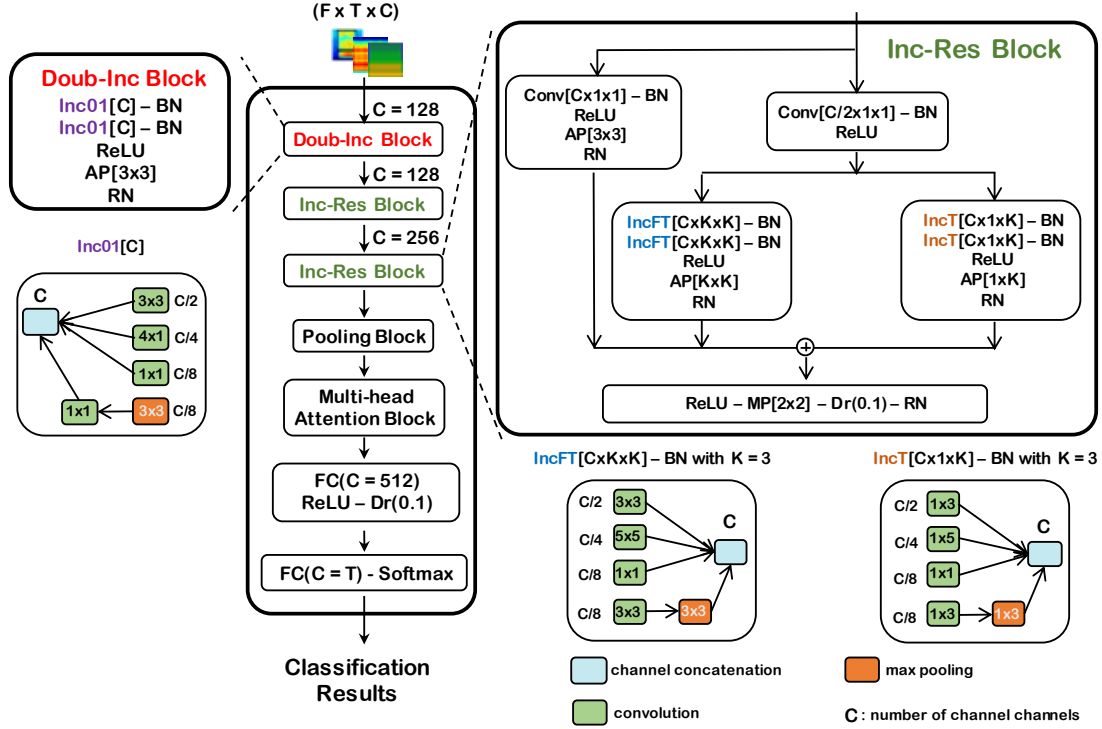


Figure 18: The proposed architecture for the Back-end classification.

to enhance the diversity of the training data and enlarge Fisher's criterion (i.e. the ratio of between-class distance to within-class variance in the feature space). Eventually, the augmented spectrograms are fed into a back-end classifier, to output the classification results.

4.4.3 Spatio-Temporal Focusing layers

The respiratory events/recordings are resampled to 4 kHz as abnormal respiratory sounds typically have frequency bands within the range of 60-2000 Hz [60]. Next, while the resampled respiratory events with different lengths are duplicated to ensure they have a consistent length of 10 seconds, audio recordings are also duplicated to make sure that all audio recordings have a consistent duration of 30 seconds. A band-pass filter of 60-2000 Hz is then applied to suppress background noise in each respiratory event/recording. Finally, these

respiratory events/recordings undergo a transformation process to generate two-dimensional spectrograms. This transformation applies CWT, which employs Amor, Bump, and Morse as the mother wavelet functions [35]. As a result of these transformations, three types of spectrograms are obtained from each respiratory event/recording. Finally, while all spectrograms at the event level are scaled to different sizes of 128×128 , 128×256 , and 128×512 , each type of spectrogram at the recording level is scaled to other non-identical sizes of 140×256 , 140×512 , 140×1024 (i.e. frequency bands $\times$ the number of time frames).

The proposed architecture for the back-end classification is illustrated in Figure 18, which is based on Inception-residual-based network architecture inspired by our prior work in [177]. In detail, our proposed network architecture comprises one Doub-Inc Block, two Inc-Res Blocks, one Pooling Block, one Multi-head Attention Block, and two fully-connected layers. The Doub-Inc Block includes two Inc01 blocks followed by Batch Normalization (BN), Rectified Linear Unit (ReLU), Average Pooling (AP[Kernel Size]), Dropout (Dr(Drop Ratio)), and Residual Normalization (RN($\lambda = 0.4$)) inspired from [170]). The Inc01 block is constructed as a variant of the naive inception layer [171] with fixed kernel sizes of $[3 \times 3]$, $[1 \times 1]$, and $[4 \times 1]$. Next, two Inc-Res Blocks share the same architecture as inspired by our prior work in [177], but channel number increase from 128 to 256 to form a deeper view of the channel dimension. The detailed structure for every Inc-Res Block is described in the right-hand side of Figure 18, which presents two main blocks of IncFT[Channel $\times$ Kernel Size] block and IncT[Channel $\times$ Kernel Size] block, and layers of Conv, BN, Dr, ReLU, AP, Max Pooling (MP[Kernel Size]), and RN($\lambda = 0.4$). Notably, IncFT[Channel $\times$ Kernel Size] and IncT[Channel $\times$ Kernel Size] layers propose different kernel sizes defined as $[K \times K]$ and $[1 \times K]$. The value of K is changed as details on the bottom part of Figure 18. The idea of IncFT[Channel $\times$ Kernel Size] is proposed to strengthen the network to learn effectively not only the widespread frequency bands but also the distribution of

energy in certain frequency bands across the temporal length of the input spectrograms. Additionally, we suppose that anomalies in lung sounds come from different duration with different changes in their frequency, which might cause more changes in the temporal dimension of each spectrogram rather than in certain frequency bands. Therefore, the IncT[Channel $\times$ Kernel Size] is proposed to focus completely on temporal information from different dimensions of the time frame in spectrogram inputs. Following each IncFT[Channel $\times$ Kernel Size] and IncT[Channel $\times$ Kernel Size] block, a ReLU, an AP layer, and RN layer are applied before adding sub-branch results together. The Pooling Block makes use of global pooling layers to extract three features from the second Inc-Res Block: (1) global average pooling across the channel dimension, (2) global max pooling across the temporal dimension, and (3) global average pooling across frequency dimensions.

Furthermore, it has been observed in [173] that individual attention heads acquire distinct sets of weight matrices. When these self-attention heads are combined, they create a multi-head self-attention layer. This layer enhances the learning performance to generate more comprehensive embedding features among different anomalies in respiratory sounds. As a result, the output of the Pooling Block is then presented to the Multi-head Attention block as shown in Figure 18. At each Attention block, we apply three multi-head attention layers on three dimensions of frequency, time, and channel. Each multi-head attention layer is configured to have 16 as the number of heads and 32 as the key dimension. The output of each multi-head attention layer is a one-dimensional embedding feature. We then concatenate these features before feeding them into fully connected blocks. While the first dense layer comprises a fully connected layer (FC[C = 512]) followed by a ReLU, Dr, the second dense layer comprises a fully connected layer (FC[C = T]) followed by a Softmax, where T is defined according to the number of target classes.

Model implementation and Training Loss functions: We implement our proposed models in this thesis using the Tensorflow framework. As we propose to use mixup data augmentation, the labels are not one-hot format. Therefore, we use Kullback-Leibler (KL) divergence loss in the proposed networks as shown in Eq. (54).

4.4.4 Results on Event Level

On Event Level (Task 1-1 and Task 1-2 on Validation set): As the experimental results are shown in Table 27, it can be seen that when the number of time frames in each type of Wavelet spectrogram is extended (i.e. from 128 to 256 and 512), the performance is further improved in both Task 1-1 and Task 1-2. This can be explained as when more temporal information is provided, the proposed model has the greater ability to exploit information effectively across the temporal length of each spectrogram. For instance, extending from 128 to 256 and 512 helps Wavelet (Bump) achieves a score improvement from 0.83 to 0.86 in Task 1-1 and an improvement of 3% in Score in Task 1-2. Compare among three evaluating spectrograms, Wavelet (Bump) outperforms Wavelet (Morse) and Wavelet (Amor), and achieves the highest Scores at the size of 128×512 in both Task 1-1 and Task 1-2 with 0.86 and 0.79, respectively. It indicates that Wavelet (Bump) spectrogram is the most appropriate representation to capture distinct features of respiratory events.

4.4.5 Results on Recording Level

On Record Level (Task 2-1 and Task 2-2 on Validation set): Similar to Task 1-1 and Task 1-2, the increase of time frames (ranging from 256 to 512 and 1024) in each type of Wavelet spectrogram results in improvements in both Task 2-1 and Task 2-2 as shown in Table 28. This again indicates the benefits of widening time frames in every spectrogram and the efficiency of our proposed

model which is developed to focus on temporal information. For instance, the significant improvement of 15% in AS and 13% in HS are obtained in Task 2-1 when Wavelet (Bump) is extended from 140×256 to 140×1024 . Notably, when the spectrograms are set to 140×1024 , Wavelet (Morse) outperforms Wavelet (Amor) and Wavelet (Bump), and achieves the highest performance of Score in both Task 2-1 (at 0.71) and Task 2-2 (at 0.55). This indicates that Wavelet (Morse) spectrogram is the most suitable representation for respiratory recordings.

On IEEE BioCAS 2023 grand challenge (All tasks): Given the experimental results on both the event level and the entire encoding level, we indicate that our models with Wavelet (Bump) at the size 128×512 and Wavelet (Morse) with the size of 140×1024 achieve the best Scores on Task 1 and Task 2, respectively. We, therefore, use these systems to submit to the IEEE BioCAS 2023 grand challenge and evaluate our systems with the blind Test set [60]. As the results shown in Table 29, our proposed model trained on Wavelet (Bump) at size 128×512 has achieved Top-1 performance in tasks of event level. In particular, we gain the highest Score of 0.810 and 0.667 on Task 1-1 and Task 1-2, respectively. As regards the recording level, when the proposed model is trained with Wavelet (Morse) spectrogram at a size of 140×1024 , we surpass other systems to have the best Score of 0.608 in Task 2-2. Even though Task 2-1 has a lower Score of 0.744, compared to that in Top-2, our proposed system still outperforms others in general.

These results demonstrate the robustness of the proposed model across both event-level and recording-level tasks.

Table 20: Performance comparison of proposed deep learning frameworks with model size.

Inception-based		
Framework	Scores (Spec./Sen./ICB.)	Model Size (MB)
Inc-01	56.3/ 40.5 /48.4	2.8
Inc-02	69.7/31.9/50.8	3.4
Inc-03	81.7/28.4/ 55.1	4.7
Inc-04	84.0 /24.8/54.4	5.5
Inc-05	80.5/26.3/53.4	9.6
Inc-06	74.8/30.0/52.4	10.7
Benchmark		
Framework	Scores (Spec./Sen./ICB.)	Model Size (MB)
VGG16	70.1/28.6/49.3	138
VGG19	69.7/28.4/49.1	143
MobileNetV1	75.5/14.3/44.9	16
MobileNetV2	74.7/16.1/45.4	14
ResNet50	88.0 /15.2/ 51.6	98
DenseNet201	71.7/30.3/51.1	80
InceptionV3	70.9/ 32.2 / 51.6	92
Xception	75.7/22.1/48.9	88
Transfer Learning		
Framework	Scores (Spec./Sen./ICB.)	Model Size (MB)
VGG14	82.1 /28.1/ 55.1	120
DaiNet19	76.4/26.9/51.7	95
MobileNetV1	64.4/ 40.3 /52.3	16
MobileNetV2	76.0/32.7/54.4	14
LeeNet24	70.7/30.9/52.8	25
Res1DNet30	74.9/26.7/50.8	30
ResNet38	71.6/32.2/51.9	80
Wavegram-CNN	69.0/38.1/53.5	35

Table 21: The MLP architecture used for training embedding features.

Setting layers	Output
FC(4096) - ReLU - Dr(10%)	4096
FC(4096) - ReLU - Dr(10%)	4096
FC(1024) - ReLU - Dr(10%)	1024
FC(4) - Softmax	4

Table 22: Performance comparison of fusion strategies of inception based and transfer learning based frameworks.

Fusion Strategies	Spec.	Sen.	ICB.
Pre-trained VGG14	82.1	28.1	55.1
Inc-03	81.7	28.4	55.1
The early fusion	79.9	30.9	55.4
The middle fusion	87.3	25.1	56.2
The late fusion	85.6	30.0	57.3

Table 23: Comparison against the state-of-the-art systems.

Method	Spec.	Sen.	ICBHI Score
HMM [149]	38.0	41.0	39.0
DT [162]	75.0	12.0	43.0
1D-CNN [163]	36.0	51.0	43.0
SVM [150]	78.0	20.0	47.0
Autoencoder [32]	69.0	30.0	49.0
ResNet [164]	69.2	31.1	50.2
ResNet [165]	63.2	41.3	52.3
Inception [151]	73.2	32.2	53.2
CNN-RNN [33]	81.0	28.0	54.0
ResNet50 [166]	72.3	40.1	56.2
Our best system	85.6	30.0	57.3

Table 24: Performance comparison on WA, GA, and WM branches on the official test set in SPRSound dataset 2022.

System	WA-branch		GA-branch		WM-branch	
	SE/SP	AS/HS	SE/SP	AS/HS	SE/SP	AS/HS
Task 1-1	77.3/80.7	78.9/78.9	87.9/72.1	80.0/79.2	74.5/88.6	81.5/80.9
Task 1-2	49.4/87.1	68.2/63.0	55.2/84.8	70.0/66.9	66.4/70.3	68.3/68.2
Task 2-1	46.5/79.1	62.8/58.6	70.3/50.6	60.5/58.8	62.4/64.4	63.4/63.4
Task 2-2	22.6/71.1	46.9/34.3	24.8/72.1	48.5/36.9	26.2/74.2	50.2/38.7

Table 25: Performance comparison on System I, II, and III on the official test set in SPRSound dataset 2022.

System	System I		System II		System III	
	SE/SP	AS/HS	SE/SP	AS/HS	SE/SP	AS/HS
Task 1-1	70.4/87.9	81.2/80.6	79.4/85.9	82.7/82.5	84.4/85.5	84.9/84.9
Task 1-2	56.1/89.5	72.8/69.0	65.3/87.7	76.5/74.9	67.8/88.3	78.1/76.7
Task 2-1	58.1/72.2	65.2/64.4	65.6/77.1	71.4/70.9	70.4/78.9	74.7/74.4
Task 2-2	19.1/91.7	55.4/31.6	25.2/83.6	54.4/38.7	36.1/80.1	58.1/49.8

Table 26: Performance of our proposed system compared to others on the official test set in SPRSound dataset 2022 (Score(%)).

Systems	Task 1-1	Task 1-2	Task 2-1	Task 2-2
Challenge Baseline [60]	75.2	61.6	56.7	37.8
Top 1 [175]	88.9	82.0	71.8	53.3
Top 2 [53]	82.0	74.3	71.1	53.1
Top 3 [176]	89.0	80.0	71.0	36.0
Our System III	84.9	77.4	74.5	53.9

Table 27: Performance comparison among spectrograms on the Validation set in Task 1-1 and Task 1-2 (event level).

System	Task 1-1			Task 1-2		
	SE/SP	AS/HS	Score	SE/SP	AS/HS	Score
Wavelet (Amor)						
128×128	0.73/0.86	0.79/0.79	0.79	0.57/0.86	0.72/0.68	0.70
128×256	0.77/0.87	0.82/0.82	0.82	0.61/0.87	0.74/0.72	0.73
128×512	0.76/0.90	0.83/0.82	0.83	0.64/0.90	0.77/0.75	0.76
Wavelet (Bump)						
128×128	0.77/0.90	0.83/0.83	0.83	0.64/0.90	0.77/0.75	0.76
128×256	0.79/0.88	0.83/0.83	0.83	0.67/0.88	0.78/0.76	0.77
128×512	0.81/0.91	0.86/0.86	0.86	0.67/0.92	0.79/0.78	0.79
Wavelet (Morse)						
128×128	0.80/0.84	0.82/0.82	0.82	0.63/0.84	0.74/0.72	0.73
128×256	0.75/0.92	0.83/0.83	0.83	0.62/0.92	0.77/0.74	0.76
128×512	0.84/0.87	0.85/0.85	0.85	0.69/0.87	0.78/0.77	0.77

Table 28: Performance comparison among spectrograms on Validation set in Task 2-1 and Task 2-2 (entire recording level).

System	Task 2-1			Task 2-2		
	SE/SP	AS/HS	Score	SE/SP	AS/HS	Score
Wavelet (Amor)						
140×256	0.66/0.59	0.63/0.62	0.62	0.33/0.59	0.46/0.43	0.45
140×512	0.56/0.78	0.67/0.65	0.66	0.35/0.78	0.56/0.48	0.52
140×1024	0.77/0.64	0.70/0.70	0.70	0.41/0.64	0.52/0.50	0.51
Wavelet (Bump)						
140×256	0.54/0.57	0.55/0.55	0.55	0.33/0.78	0.55/0.46	0.51
140×512	0.54/0.83	0.69/0.65	0.67	0.32/0.83	0.58/0.46	0.52
140×1024	0.58/0.83	0.70/0.68	0.69	0.32/0.85	0.59/0.46	0.53
Wavelet (Morse)						
140×256	0.60/0.76	0.68/0.67	0.67	0.27/0.76	0.52/0.40	0.46
140×512	0.57/0.76	0.66/0.65	0.66	0.35/0.76	0.56/0.48	0.52
140×1024	0.67/0.76	0.71/0.71	0.71	0.40/0.76	0.52/0.52	0.55

Table 29: Compare our proposed systems to the others submitted IEEE BioCAS 2023 challenge (Score(%) on the blind Test set).

Systems	Task 1-1	Task 1-2	Task 2-1	Task 2-2
Top 1 (Our system)	0.810	0.667	0.744	0.608
Top 2	0.733	0.646	0.759	0.538
Top 3	0.769	0.632	0.661	0.512
Top 4	0.720	0.593	0.665	0.549
Top 5	0.668	0.555	0.723	0.524
Top 6	0.748	0.599	0.699	0.411
Top 7	0.785	0.648	0.547	0.417
Top 8	0.756	0.467	0.658	0.458

Chapter 5

Cross Domain Adaptation

5.1 Audio-Based Deep Learning Frameworks for Detecting COVID-19

This section 5.1 evaluates a wide range of audio-based deep learning frameworks applied to breathing, cough, and speech sounds for COVID-19 detection. In general, audio recordings are first transformed into low-level spectrogram representations, including Log-Mel, Gammatone, and Wavelet-based spectrograms as defined in Chapter 3. To investigate cross-domain application from mainstream audio tasks to downstream tasks in CRAA, these representations are then fed into pre-trained deep learning models to extract high-level embedding features. Subsequently, the dimensionality of the extracted embeddings is reduced before fine-tuning, with Light Gradient Boosting Machine (LightGBM) employed as the back-end classifier.

5.1.1 Coswara Dataset and Task Defined

In this thesis, we evaluate the dataset derived from the Second DiCOVA Challenge [178]. This dataset provides audio recordings of three different sound categories: breathing, cough, speech which were collected from both COVID-19 positive and negative patients in the age group of 15 to 45 years old. For each sound category, there are a total of 1436 acoustic samples which are then divided into the Development set (965 subjects) and the Test set (471 subjects) for training and testing processes, respectively. Notably, the development set for each category includes recordings from 172 COVID19 positive individuals, while 793 recordings are from COVID-19 negative individuals, resulting in a class imbalance. To compare our proposed systems with the state-of-the-art, we follow the Second DiCOVA Challenge [178]. Particularly, we aim to detect COVID-19 positive subjects by exploring only breathing sounds, only cough sounds, only speech sounds, and using all sound categories, which match the Track-1, Track-2, Track-3, and Track-4, respectively in the Second DiCOVA Challenge. Regarding the evaluation metrics, we also obey the Second DiCOVA Challenge which uses the Area Under the Curve (AUC) and the specificity (SP) & sensitivity (SE). As SP. has to be equal to or greater than 95.13% which is an essential requirement of the challenge, a decision threshold from 0 to 1 with a step size of 0.0001 is evaluated to obtain SE. when SP. value is satisfied. As a result, we report AUC and SE. in this thesis (i.e. The SP. is always equal to or greater than 95.13% to meet the challenge’s requirement as presented in the leader board of the challenge [179]).

5.1.2 The Proposed Framework

To explore DiCOVA dataset [178], we first present a high-level architecture of the proposed framework as shown in Figure 19. Generally, an entire framework proposed is divided into three main steps: Low-level spectrogram feature extraction, high-level embedding feature extraction, and back-end classification.

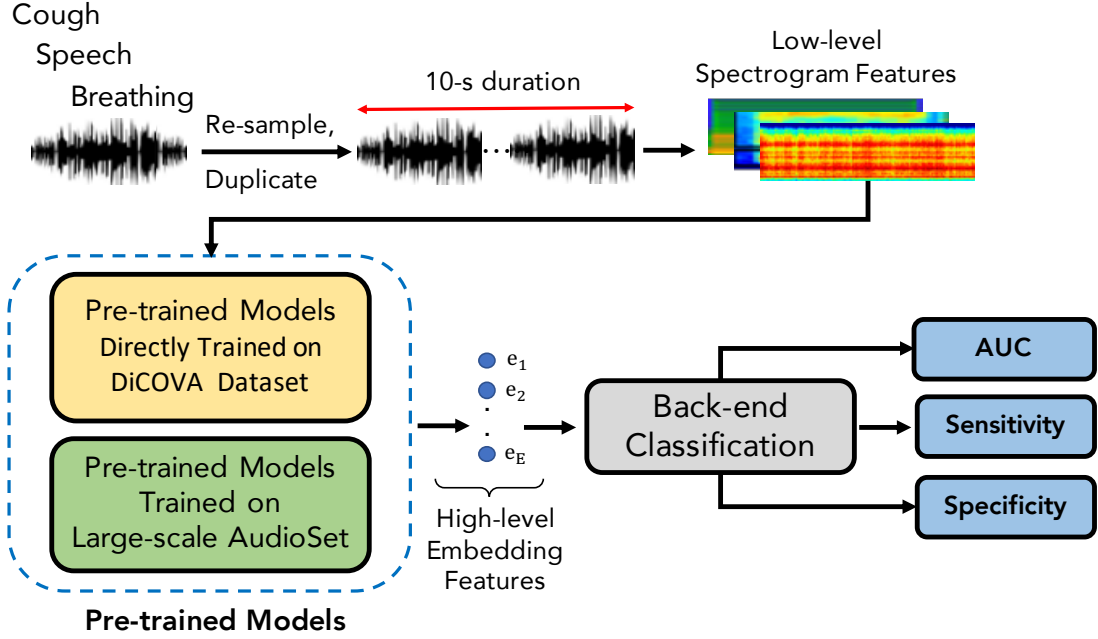


Figure 19: The high-level architecture of the proposed frameworks.

5.1.3 The Low-level Spectrogram Feature Extraction

At the first step shown in the upper stream in Figure 19, the raw audio recordings are re-sampled to 16 kHz, duplicated to ensure that all audio recordings have an equal duration of 10 seconds. These audio recordings are then transformed into spectrograms where both temporal and spectral features are presented.

5.1.4 The High-level Embedding Feature Extraction

The low-level spectrogram features are then fed into pre-trained deep learning models to extract embeddings (e.g. vectors), referred to as high-level features. The pre-trained deep learning models used for extracting high-level embedding features come from two different approaches: **(I)** pre-trained models directly trained on DiCOVA dataset and **(II)** pre-trained models trained with the large-scale AudioSet dataset [180].

(I) In the first approach, we construct deep learning models and directly

train these models on DiCOVA dataset [178] mentioned in Section 5.1.1. After the training process, we reuse these models to extract the feature map at the global pooling layer, which are considered as the high-level embedding features mentioned. As the performance of high-level embedding features depends on the low-level spectrograms and the model architectures in this approach, we therefore evaluate: three types of spectrograms such as log-Mel [66], Gammatonegram [167], and Scalogram [141]; and a wide range of benchmark deep learning models from low footprint models such as Lenet based network [181] to high-complex architectures such as Xception [182], InceptionV3 [183], etc.

To identify which spectrogram features are well-represented for breathing, cough, or speech audio inputs, we evaluate three proposed spectrograms using a Lenet based network architecture as shown in Table 30. To make sure that the spectrograms that are fed into the Lenet based model have the same size, we use the same setting parameters with window size = 2048, hop size = 1024, and filter number = 128; to generate the same spectrogram of 128×154 . Regarding the Lenet based architecture as shown in Table 30, it includes the convolutional layer (Conv [kernel size]), batch normalization (BN) [184], rectified linear units (ReLU) [157], average pooling (AP), global average pooling (GAP), dropout [158] (Dr(percentage)), fully connected (FC) and Softmax layers. Since certain low-level spectrograms are suitable for various types of audio inputs (i.e. breathing, cough, or speech), we therefore evaluate these spectrograms with different benchmark neural network architectures of VGG16 [185], VGG19 [185], MobileNetv1 [186], ResNet50 [187], Xception [182], InceptionV3 [183], and DenseNet121 [188] to achieve the best framework configuration (i.e. which low-level spectrogram and training network architecture). As evaluated network architectures are reused from the Keras library [159], the final fully connected layer of these networks is modified from 1000 (i.e. the number of classes of image object in ImageNet dataset) to 2 which matches the number of classes in DiCOVA dataset. It is worth

Table 30: Configuration of Lenet based architecture.

Lenet architecture layers	Output
Input layer (spectrogram patch of $128 \times 154 \times 1$)	
BN - Conv $[3 \times 3]$ @ 32 - ReLU - BN - AP $[2 \times 2]$ - Dr (20%)	$64 \times 77 \times 32$
BN - Conv $[3 \times 3]$ @ 64 - ReLU - BN - AP $[2 \times 2]$ - Dr (25%)	$32 \times 38 \times 64$
BN - Conv $[3 \times 3]$ @ 128 - ReLU - BN - AP $[2 \times 2]$ - Dr (30%)	$16 \times 19 \times 128$
BN - Conv $[3 \times 3]$ @ 256 - ReLU - BN - GAP - Dr (35%)	256
FC - ReLU - Dr (40%)	1024
FC - ReLU - Dr (40%)	1024
FC - Softmax	C=2

noting that we only reuse the network architectures from the Keras library [159] instead of using available weights trained with ImageNet dataset, all trainable parameters of these networks are initialized with mean and variance set to 0 and 0.1, respectively.

(II) In the second approach, we leverage three available pre-trained models trained with the large-scale AudioSet dataset in advance: PANN [152], OpenL3 [189], TRILL [190]. We evaluate whether these up-stream pre-trained models are beneficial for the downstream task of detecting COVID-19 positive in DiCOVA dataset. While PANN [152] and OpenL3 [189] leverage VGGish architecture and cross-entropy loss to train the large-scale Audioset dataset, TRILL [190] is based on Resnet and triplet loss. Additionally, while PANN [152] was trained using input signal with 10-second duration, both TRILL [190] and OpenL3 [189] analyses on short segment of 1 second. As using different network architectures, loss functions, as well as analysing different audio durations, these proposed pre-trained models may differently perform on three types of audio inputs (i.e. breathing, cough, and speech) from the DiCOVA dataset.

Regarding the high-level embedding features extracted from these pre-trained models, while OpenL3 [189] and PANN [152] extract the feature map at the global pooling layer, TRILL [190] extracts the feature map at the final layer which proves effective for different down-stream tasks mentioned in [190]. Notably, since PANN

Table 31: Comparison of low-level spectrograms (AUC/SE).

Spec	Cough	Speech	Breathing	All
GAM	70.11/10.00	73.41/21.67	73.29/15.00	77.38/25.00
log-MEL	71.39/16.66	73.15/18.33	74.31/18.33	80.11/33.33
Scalogram	63.71/8.33	68.68/10.00	78.20/20.00	69.83/18.33

pre-trained model works on 10-second input duration from AudioSet dataset that matches the input duration of our proposed system in Figure 19, only one embedding feature (e.g. one vector) is extracted from each 10-second audio recordings of the input. Meanwhile, as OpenL3 and TRILL pre-trained models were trained with 1-second audio segment of AudioSet dataset, therefore, multiple embeddings are obtained when we feed one 10-second DiCOVA audio sample into these two pre-trained models. Therefore, an average of these multiple embeddings across the time dimension is computed to obtain one embedding feature which represents for each 10-second duration audio input. Additionally, only low-level log-Mel spectrogram is used in this approach as these three pre-trained models explore this type of spectrogram for training on the large-scale AudioSet dataset.

As we present two different approaches of using pre-trained models for extracting high-level embedding features, we now refer to two main frameworks as: (I) three low-level spectrograms (log-Mel, Gammatonegram, Scalogram), pre-trained models directly trained on DiCOVA dataset, LightGBM back-end classification; and (II) low-level log-Mel spectrogram, pre-trained models trained on the large-scale AudioSet, and LightGBM back-end classification.

5.1.5 Back-end Classification

In this thesis, we use Light Gradient Boosting Machine (LightGBM) [191] as the final back-end classification model to fine-tune high-level embedding features. The LightGBM is implemented by using an available toolkit [191] and the parameters

are set to: `learning_rate = 0.02`, `objective = 'binary'`, `metric = 'auc'`, `subsample = 0.68`, `colsample_bytree = 0.28`, `early_stopping_rounds = 1000`, `num_iterations = 10000`, `subsample_freq = 1`. As we participated in all tracks in the Second DiCOVA challenge, which suggests using all audio input data (i.e. breathing, cough, speech), high-level embedding features extracted from different types of audio inputs are concatenated before feeding into the back-end LightGBM for classification.

5.1.6 Results and Performance

In the first approach mentioned in Section 5.1.4, we construct deep learning models using Tensorflow [192] and directly train these models on DiCOVA dataset with Cross-entropy loss and Adam optimizer [160].

Regarding the second approach mentioned in Section 5.1.4, we use Pytorch framework [193] to extract high-level features from the pre-trained PANN model as this network was built in a Pytorch environment. Meanwhile, both the pre-trained TRILL and OpenL3 models are built in Tensorflow for extracting high-level embedding features.

Performance comparison of frameworks (I): three low-level spectrograms, pre-trained models directly trained on DiCOVA dataset, and LightGBM back-end classification

We first evaluate how low-level spectrograms affect the performance in frameworks (I). As the results are shown in Table 31, log-Mel performs better than Gammatonegram and Scalogram with the highest scores of 71.39/16.66 in cough while still achieves the second-highest scores of 73.15/18.33 and 74.31/18.33 in speech and breathing, respectively. As a result, we only use log-Mel spectrogram for evaluating network architectures.

Table 32: Comparison of pre-trained deep learning network architectures directly trained on DiCOVA dataset (AUC/SE).

Input	Cough	Speech	Breathing	All
VGG16	70.05/10.00	75.64/10.00	71.75/16.66	78.17/30.00
VGG19	63.07/5.00	67.81/8.33	70.28/15.00	68.24/6.67
MobileNetv1	58.92/5.00	66.86/16.67	67.76/13.33	70.73/25.00
ResNet50	66.67/10.00	70.07/21.67	69.39/18.33	77.35/23.33
Xception	71.21/31.66	70.91/28.33	72.75/28.33	80.81/43.33
InceptionV3	61.97/8.33	74.39/30.00	67.56/18.33	78.83/30.00
DenseNet121	68.12/23.33	75.55/20.00	72.63/18.33	80.10/31.66

As Table 32 shows, it can be seen that Xception outperforms the other network architectures. However, the high-complex Xception model only slightly improves, compared to the low footprint LeNet based architecture. Notably, when high-level embedding features extracted from breathing, cough, and speech inputs by Xception model are concatenated to detect COVID-19 (i.e. Track-4 in the Second DiCOVA Challenge), it helps to significantly improve the performance. This achieves the score of 80.81/43.33 compared with 71.21/31.66, 70.91/28.33, 72.75/28.33 for only cough, speech, and breathing, respectively.

Performance comparison of frameworks (II): low-level log-Mel spectrogram, pre-trained models trained on AudioSet dataset, and LightGBM back-end classification

As Table 33 shows, it can be seen that different pre-trained models work well on different types of sound input. Particularly, the best scores of 79.82/48.33 are associated with TRILL for cough sounds input. PANN achieves the best scores of 84.05/55.00 for breathing. Meanwhile, speech sounds input yields the best scores of 86.65/61.67 with OpenL3. Notably, using pre-trained models with AudioSet dataset outperforms the frameworks (I) analysed in Section 5.1.6 in terms of detecting COVID-19 with only breathing, cough, or speech.

Comparing the performance of various models among audio inputs, breathing

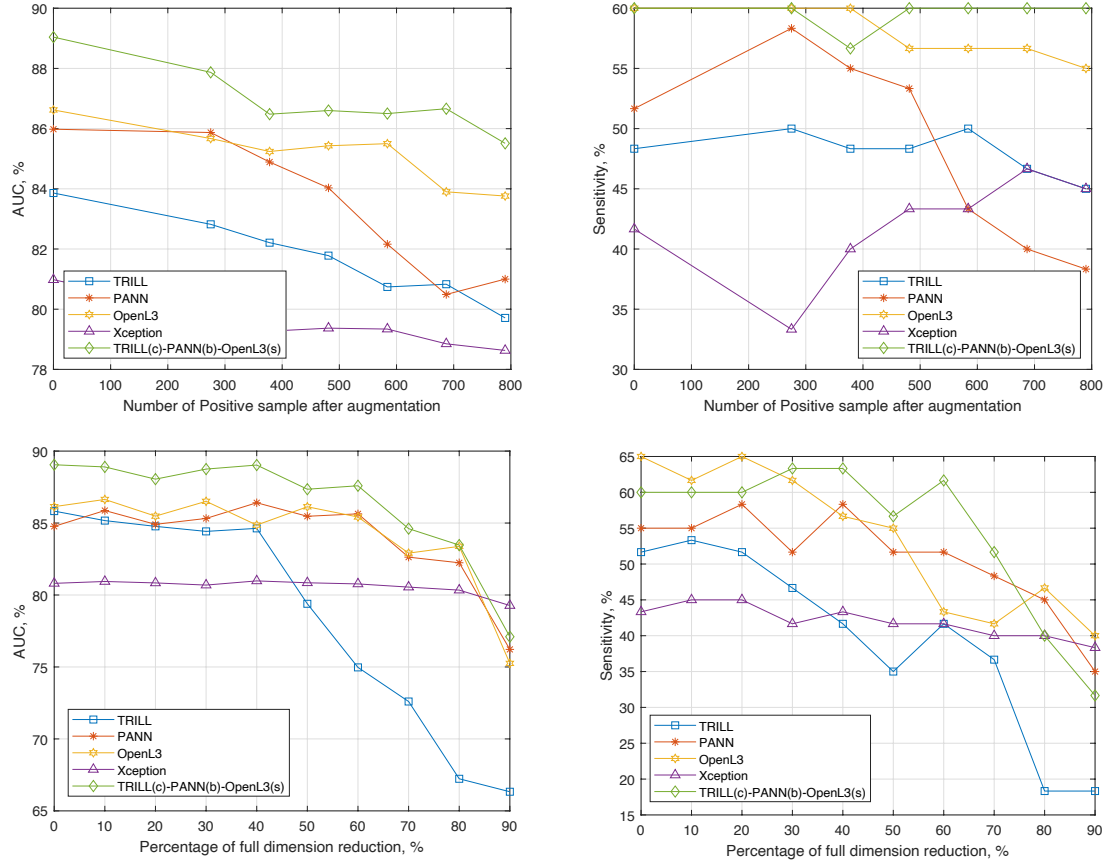


Figure 20: Influence of over-sampling on positive samples and dimensionality reduction within Track-4 DiCOVA Challenge.

and speech show greater potential to detect COVID-19 rather than cough. However, when we concatenate high-level embedding features extracted from breathing, cough, and speech for Track-4 of the challenge, the performance from TRILL shows a significant improvement reaching 85.83/51.67. Meanwhile, OpenL3 demonstrates the highest scores of 86.14/65.00 compared to the others in this track.

Influence of over-sampling on positive samples and dimensionality reduction of high-level features

As the proposed frameworks with a concatenation of high-level features prove effective to detect COVID-19 in Track-4 of the Second DiCOVA Challenge, we further conduct experiments to evaluate how over-sampling on positive samples and

Table 33: Comparison of pre-trained models trained on AudioSet dataset (AUC/SE).

Pre-trained model	Cough	Speech	Breathing	All
PANN	70.49/26.66	79.33/33.33	84.05/55.00	84.77/55.00
TRILL	79.82/48.33	80.50/46.67	76.97/33.33	85.83/51.67
OpenL3	78.42/38.33	86.65/61.67	82.40/50.00	86.14/65.00

reducing dimension over high-level features affect the performance. In these experiments, we concatenate (1) three TRILL based high-level features, (2) three PANN based high-level features, (3) three OpenL3 based high-level features, (4) three Xception based high-level features, and (5) (TRILL(c)-PANN(b)-OpenL3(s)) high-level features for cough, breathing, and speech, respectively. We conduct the combination among high-level features (TRILL(c)-PANN(b)-OpenL3(s)) as each feature shows effective for each audio data input as shown in Table 33.

To identify less significant dimensions of high-level features, we firstly compute an average of each dimension from all features associated with positive and negative COVID-19 independently. Then, two vectors representing two groups of COVID-19 positive and negative features are obtained. Based on the absolute difference of these two vectors from each dimension (i.e. the lower absolute difference is related to the less significant dimension), the dimension of the feature set was reduced from 10% to 90%. Note that the dimension of high-level features is reduced before concatenating. To over-sample the positive cases, we apply SVM-SMOTE [194] on the high-level embedding features, then increase the the positive samples from double to five times.

As Figure 20 shows, oversampling on positive samples decrease the AUC scores. This method only helps to improve SE. scores with PANN (double of positive samples) and Xception (five times of positive samples). Meanwhile, when we reduce the dimension of high-level embedding features from 20 to 40%, it almost helps to improve both AUC and SE. scores. As a result, we can achieve the best scores: 80.32/48.33 with TRILL (at the reducing dimension of 20%) for

cough input only, 84.05/55.00 with PANN for breathing input only (no reduction), 86.65/61.67 with OpenL3 for speech only (no reduction), and 89.03/63.33 with (TRILL(c)-PANN(b)-OpenL3(s)) (at the dimensionality reduction of 40%) for all audio inputs.

Performance comparison among different back-end classifiers

We further compare the LightGBM with other machine learning models implemented using Scikit-Learn toolkit [195] on Support Vector Machine (SVM with $C=1.0$, Kernel='RBF'), Random Forest (RF with Max Depth of Tree = 20, Number of Trees = 100), Multi-layer perceptron (MLP with 4096 nodes, Adam optimization, Max iter = 200, Learning rate = 0.001, Entropy Loss), and Linear Regression (LR with $C=1.0$). Noting that we keep TRILL based embedding (at the dimensionality reduction of 20%) for cough, PANN and OpenL3 based embeddings (without reduction) for breathing and speech, respectively. Similarly, we continue to apply the best concatenation of (TRILL(c)-PANN(b)-OpenL3(s)) (with a dimensionality reduction of 40%) for all audio inputs. As a result, the comparison in Table 36 again shows that LightGBM model consistently achieves the best scores, outperforming other models.

Performance comparison across the top-10 systems submitted for the Second DiCOVA Challenge

As performance comparison to the state-of-the-art systems [179] is shown in Table 34 and Table 35, we can achieve the top-6 in Track-1 with breathing input only and the top-3 in Track-2 with cough input only. Notably, our proposed systems outperform the state-of-the-art, achieve the top-1 in both Track-3 and Track-4 with speech input only and all audio inputs, respectively.

To evaluate the optimal score of the LightGBM with the best-selected embedding features for cough, speech, breathing and all types of audio inputs, we conducted ten times of running the experiments on a different randomly chosen

Table 34: Performance comparison with the state-of-the-art systems on Test set (Cough and Speech).

System	Cough				Speech			
	AUC	SE.	Precision	F1 score	AUC	SE.	Precision	F1 score
1st system	81.97	36.67	52.38	43.14	85.21	45.00	57.45	50.47
2nd system	81.21	48.33	59.18	53.21	84.73	38.33	53.49	44.66
3rd system	80.12	35.00	51.22	41.58	84.55	51.67	60.78	55.86
4th system	79.06	35.00	51.22	41.58	84.26	43.33	56.52	49.05
5th system	77.85	46.67	58.33	51.85	84.04	48.33	59.18	53.21
6th system	77.60	33.33	50.00	40.00	82.98	43.33	56.52	49.05
7th system	76.98	25.00	42.86	31.58	82.83	38.33	53.49	44.66
8th system	76.36	30.00	47.37	36.74	81.95	48.33	59.18	53.21
9th system	75.95	28.33	45.95	35.05	81.86	31.67	48.72	38.39
10th system	75.71	35.00	51.22	41.58	80.92	45.00	57.45	50.47
Baseline system	74.89	36.67	52.38	43.14	84.26	43.33	56.52	49.05
Our system	80.32	48.33	59.18	53.21	86.65	61.67	64.91	63.25

Table 35: Performance comparison with the state-of-the-art systems on Test set (Breathing and All).

System	Breathing				All			
	AUC	SE.	Precision	F1 score	AUC	SE.	Precision	F1 score
1st system	87.18	48.33	59.18	53.21	88.44	58.33	63.64	60.87
2nd system	86.72	40.00	54.55	46.16	87.26	63.33	65.52	64.41
3rd system	86.41	45.00	57.45	50.47	86.87	58.33	63.64	60.87
4th system	85.77	41.67	55.56	47.62	85.79	33.33	50.00	40.00
5th system	84.50	31.67	48.72	38.39	85.37	60.00	64.29	62.07
6th system	82.16	41.67	55.56	47.62	84.70	55.00	62.26	58.41
7th system	82.05	41.67	55.56	47.62	84.26	38.33	53.49	44.66
8th system	80.84	40.00	54.55	46.16	83.78	46.67	58.33	51.85
9th system	80.55	30.00	47.37	36.74	80.51	40.00	54.55	46.16
10th system	80.35	36.67	52.38	43.14	74.15	35.00	51.22	41.58
Baseline system	84.50	31.67	48.72	38.39	84.70	55.00	62.26	58.41
Our system	84.05	55.00	62.26	58.41	89.03	63.33	65.52	64.41

Test set. Next, we calculated an average confidence interval (CI) of [0.76, 0.83] in cough, [0.83, 0.89] in speech, [0.81, 0.87] in breathing, and [0.85, 0.91] in all audio inputs. All of them match with the mentioned AUC scores of 80.32, 86.65, 84.05, and 89.03 for cough, speech, breathing, and all, respectively.

Table 36: Performance comparison across different back-end classification models on Test set (AUC/SE).

Model	Cough	Speech	Breathing	All
LR	76.22/35.00	80.90/40.00	78.61/43.33	77.29/43.33
SVM	78.40/41.66	85.93/53.33	82.25/51.66	85.73/55.00
RF	78.78/43.33	81.06/48.33	79.06/38.33	85.34/51.66
MLP	74.80/41.66	81.53/56.66	80.89/41.66	83.75/46.67
LightGBM	80.32/48.33	86.65/61.67	84.05/55.00	89.03/63.33

5.2 An Efficient Embedding Alignment Framework for Detecting Respiratory Anomalies Using Deep Learning

We introduce an efficient embedding-based domain adaptation approach for improving diagnostic accuracy across benchmark respiratory sound datasets with diverse age groups and patient populations (ICBHI 2017, HF_lung_v1 2022, and SPRSound 2023). Our method leverages continuous wavelet transformation to extract Wavelet-based spectrograms, enabling multi-resolution analysis and improved discrimination between normal and abnormal respiratory sounds. Furthermore, we propose an embedding-based alignment technique that creates a unified feature space for source and target domains, facilitating the alignment of class-wise distributions and improving cross-domain detection accuracy. Unlike traditional fine-tuning methods, which often suffer from limitations such as forgetting the source domain, our approach conduct the first work on consistent alignment and generalization across various datasets.

5.2.1 Front-End Feature Extraction

First, the respiratory events are resampled to 4 kHz since abnormal respiratory sounds typically occupy the range of 60-2000 Hz [60]. Next, the resampled respiratory events with different lengths are duplicated to ensure they have a consistent length of 10 seconds. A band-pass filter of 60-2000 Hz is then applied to suppress background noise in each respiratory event. Finally, these respiratory events $y(t)$ undergo a transformation process to generate two-dimensional scalograms. This transformation applies Continuous Wavelet transformation, which employs Amor, Bump, and Morse as the Wavelet mother functions [141, 142, 143].

5.2.2 Deep Learning Backbone Architecture

The proposed back-end classification architecture, illustrated in Figure 21, is based on our prior work on Inception-residual-based networks [29, 35]. This architecture comprises a Doub-Inc Block, two Inc-Res Blocks, a Pooling Block, and a Multi-head Attention Block. The Doub-Inc Block consists of two Inc01 blocks, followed by a batch normalization, an average pooling layer, and an activation layer. The Inc01 block is established as a variant of the traditional Inception layer [171], utilizes fixed kernel sizes of $[3 \times 3]$, $[1 \times 1]$, and $[4 \times 1]$. By incorporating these components, we aim to effectively extract and represent relevant features.

Next, two identical Inc-Res Blocks are implemented to enhance the capacity to capture channel-wise features. The channel size is doubled from 128 to 256 in these blocks. As illustrated on the right-hand side of Figure 21, each Inc-Res Block comprises three fundamental building blocks: IncF[Channel $\times$ Kernel Size] block, IncFT[Channel $\times$ Kernel Size] block, and IncT[Channel $\times$ Kernel Size] block. These blocks incorporate distinct kernel sizes including $[K \times 1]$, $[K \times K]$, and $[1 \times K]$, with K fixed at 3. Following each of these blocks, an activation function (ReLU), an average pooling layer (AP), a max pooling layer (MP), a batch normalization (BN) and a residual normalization(RN($\lambda = 0.4$)) [170] layer are sequentially

applied to refine the feature representations.

We observed in Figure 9 that each Scalogram can provide different information of frequency and timing by the distinct characteristic of every mother wavelet function. Therefore, this led to the idea of using three main blocks IncF, IncT, and IncFT to strengthen the network to learn effectively not only the widespread frequency bands but also the distribution of energy in certain frequency bands across the temporal length of the input scalograms. First, anomalies in lung sounds induce several distinct changes in their frequency domain, potentially causing more variations in certain frequency bands rather than in the temporal dimension of each scalogram. Consequently, the IncF[Channel $\times$ Kernel Size] is designed to focus exclusively on different parts of the frequency spectrum in scalogram inputs. Second, different lengths of respiratory events produce more variations in the temporal dimension of each scalogram rather than in specific frequency bands. As a result, the IncT[Channel $\times$ Kernel Size] is designed to focus solely on temporal information across different dimensions of the time frame in scalogram inputs. Third, we also hypothesized that anomalies in lung sounds arise from various durations with distinct changes in their frequencies, potentially causing more variations in both frequency and temporal dimensions of each scalogram. Therefore, the IncFT[Channel $\times$ Kernel Size] is designed to comprehensively focus on both frequency and temporal information from different dimensions in each scalogram.

After Inc-Res Blocks, we propose a Pooling Block which makes use of global pooling layers to extract three features from the second Inc-Res Block: (1) global average pooling across the channel dimension, (2) global max pooling across the temporal dimension, and (3) global average pooling across frequency dimensions. Furthermore, as demonstrated in [173], individual attention heads within a multi-head self-attention layer learn distinct sets of weights, enabling the model to capture diverse relationships between anomaly patterns in respiratory sounds. To exploit this capability, the output of the Pooling Block is fed into a Multi-head

Attention Block, as shown in Figure 21. This block comprises three multi-head attention layers, each operating on a specific dimension: frequency, time, and channel. Each layer is configured with 16 attention heads and a key dimension of 32. The output of each multi-head attention layer is a one-dimensional embedding feature (i.e. the embedding feature is described as a vector $\mathbf{e} = [e_1, e_2, \dots, e_E]$ in Figure 21.

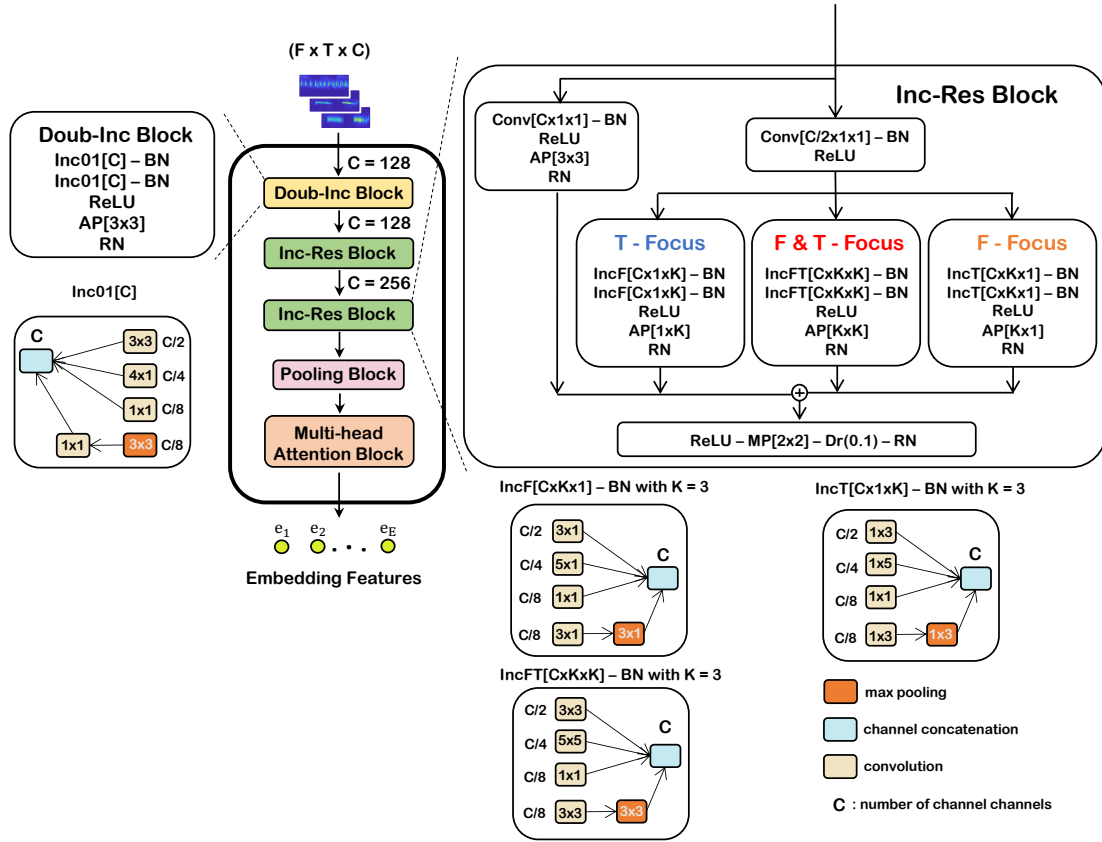


Figure 21: The detailed architecture of deep learning backbone architecture.

5.2.3 Proposed Feature Alignment Framework

The proposed feature alignment framework is shown in Figure 22. It has two main streams for the source and target domains. Note that the parameters of the two streams are shared. In each stream, there are three sub-branched networks where

each branch is based on the backbone architecture to learn from each type of scalogram. We use three types of scalograms and one branch uses each scalogram as its input. This is because different scalograms can capture different acoustic features and characteristics as shown in Figure 9 and Table. 1.

As details are presented in Figure 22, either source domain stream or target domain stream, the embedding features from every Attention Block of each Scalogram are finally fed into the following individual Deep Neural Network (DNN)-based block for classification in each sub-branch. We note that DNN_WA, DNN_WM, or DNN_WB blocks are DNN blocks for Wavelet Amor, Wavelet Morse, and Wavelet Bump, respectively. These DNN-based blocks share the same network architecture, which comprises two dense layers. The first dense layer comprises a fully connected layer (FC[L = E]) followed by BN, ReLU, and Dropout (Dr), where E is the dimensional length of the embedding feature. Meanwhile, the second dense layer comprises a fully connected layer (FC[L = 3]) followed by a Softmax, where L is defined according to the number of target classes (e.g, Wheeze, Crackle, and Normal).

Combiner: This block, as shown in the lower part of Figure22 comprises two main blocks: Combiner and DNN_Comb. The Combiner block gathers embedding features: $\mathbf{e}_{WA} = [e_1, e_2, \dots, e_{WA}]$, $\mathbf{e}_{WM} = [e_1, e_2, \dots, e_{WM}]$, $\mathbf{e}_{WB} = [e_1, e_2, \dots, e_{WB}]$ outputted from Attention blocks in each branch to obtain a Combined-feature $\mathbf{a} = [a_1, a_2, \dots, a_E]$. Since each scalogram contains distinct features, we propose this Combiner block because the Combined-feature from this Combiner block is potentially more effective in representing features from all the scalograms for detecting anomalies in respiratory sounds. In this work, we assume that the feature embeddings $\mathbf{e} = [e_1, e_2, \dots, e_E]$ from each scalogram has a linear relationship across their dimensions. We then derive a linear combination method. This linear combination is proposed to combine the three high-level features with trainable weight and bias:

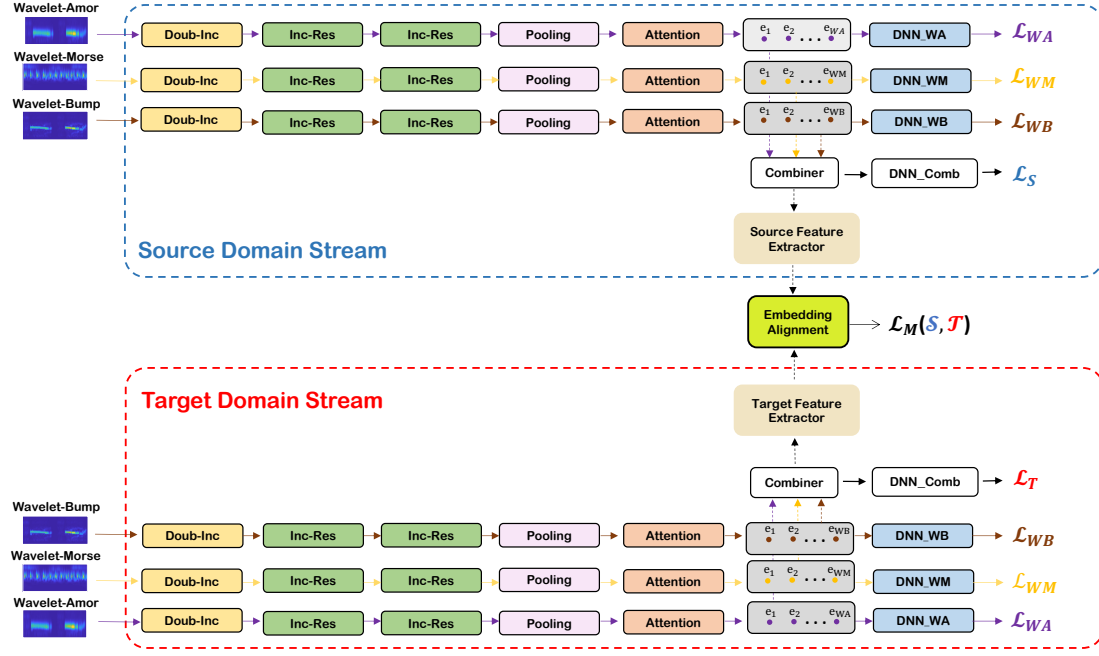


Figure 22: The Feature Alignment Framework with two main Source and Target streams.

$$\mathbf{a} = \text{ReLU}(\mathbf{e}_{WM}\mathbf{w}_{WM} + \mathbf{e}_{WA}\mathbf{w}_{WA} + \mathbf{e}_{WB}\mathbf{w}_{WB} + \mathbf{w}_{bias}), \quad (51)$$

where $\mathbf{w}_{WM/WA/WB/bias}[w_1, w_2, \dots, w_E]$ are the trained parameters. The Combined feature $\mathbf{a}$ is then fed into the DNN_Comb block for classification. The DNN_Comb block presents the same architecture as DNN_WM, DNN_WA, or DNN_WB blocks.

Next, **Source feature extractor** and **Target feature extractor** are implemented to extract Combined-feature $\mathbf{a} = [a_1, a_2, \dots, a_E]$ in each stream for embedding-based alignment method.

Training loss functions: We propose different loss functions to train the proposed system. In particular, we use Kullback-Leibler (KL) loss for each branch to learn from individual scalogram. When using mix-up data augmentation [168], the labels are not in one-hot format. We set the learning rate to 0.0001 and

applied Adam optimizer [76]. KL-loss functions of L_{WA} , L_{GA} , L_{WM} are used to classify target classes in each individual scalogram branch.

$$\mathcal{L}_S = \alpha_S(\mathcal{L}_{WM} + \mathcal{L}_{WA} + \mathcal{L}_{WB}) + \mathcal{L}_{Comb} \quad (52)$$

$$\mathcal{L}_T = \alpha_T(\mathcal{L}_{WM} + \mathcal{L}_{WA} + \mathcal{L}_{WB}) + \mathcal{L}_{Comb} \quad (53)$$

$$\mathcal{L}_M(\mathcal{S}, \mathcal{T}) = \sum_{i=1}^N (\mathcal{S}_i - \mathcal{T}_i)^2 / N, \quad (54)$$

where $\mathcal{L}_S$ and $\mathcal{L}_T$ are the final loss function of Source domain stream and Target domain stream, respectively. The values of loss weight ratios of α_S and α_T and γ is set to 1/3.

Feature Alignment: The embedding-based alignment method, as illustrated in Figure 22, consists of two steps: Initialization and Adaptation.

(i) Initialization: In this step, two mainstream network architectures are initialized with the pre-trained network weights on the source domain, (ii) Adaptation: In this step, two parallel networks are adapted where both networks are trained on both source and target modalities. In detail, source and target modalities are coming from a pair of sample sharing same class. (e.g. a sample in the target stream has the corresponding label with sample in the source stream).

The source stream networks consisting of Source feature extractor in Figure 22, are further trained on the source modality. The target stream including Target feature extractor, is fine-tuned on the target modality. Both the source and target modality networks get separately trained to classify Wheeze, Crackle, and Normal. At the same time, their feature extractors are trained to minimize the Mean Squared Error (MSE) between extracted features $\mathcal{S}$ and $\mathcal{T}$ of corresponding sample of the source modality and target modality, respectively. The training

of the combination of both the source modality and target modality networks is performed using the following loss function:

$$\mathcal{L} = (1 - \alpha_M)(\mathcal{L}_S + \mathcal{L}_T) + \alpha_M \mathcal{L}_M(\mathcal{S}, \mathcal{T}), \quad (55)$$

where α_M represents different loss weight values.

5.2.4 Experimental Setup and Results

To address the problem of imbalanced data discussed in Section 2.5, we propose three data augmentation techniques after scalograms are generated during the feature extraction step. First, we randomly oversample the scalograms to ensure an equal number of scalograms per class in each batch size. This helps to balance the representation of different classes in the training data. Second, we apply SpecAugment [169] as every scalogram within each batch are then randomly cropped with a reduction of 10 bins in both the time and frequency dimensions. This encourages the learning process to focus on the partial loss of information at each dimension. Third, we apply the mix-up data augmentation technique [168] to increase the variation in the training data. This technique combines pairs of scalograms from different classes, creating new synthetic samples. The goal is to enhance the diversity of the training data and enlarge Fisher's criterion (i.e. the ratio of between-class distance to within-class variance in the feature space) [196]. Let's consider two original scalograms as $\mathbf{X}_1$, $\mathbf{X}_2$ and expected labels as $\mathbf{y}_1$, $\mathbf{y}_2$, new scalogram patches are generated as below equations:

$$\mathbf{X}_{\text{mp1}} = \mathbf{X}_1\gamma + \mathbf{X}_2(1 - \gamma) \quad (56)$$

$$\mathbf{X}_{\text{mp2}} = \mathbf{X}_1(1 - \gamma) + \mathbf{X}_2\gamma \quad (57)$$

$$\mathbf{y}_{\text{mp1}} = \mathbf{y}_1\gamma + \mathbf{y}_2(1 - \gamma) \quad (58)$$

$$\mathbf{y}_{\text{mp2}} = \mathbf{y}_1(1 - \gamma) + \mathbf{y}_2\gamma \quad (59)$$

where γ is random coefficient from *Beta* distribution, $\mathbf{X}_{\text{mp1}}$, $\mathbf{X}_{\text{mp2}}$ and $\mathbf{y}_{\text{mp1}}$, $\mathbf{y}_{\text{mp2}}$ are new scalogram patches and generated labels, respectively.

Training Data Partition

Table 37: Training sample distribution across source and target datasets. For target datasets, values are reported as source $\rightarrow$ target.

Source Training Dataset			
SPRSound 2023 (Official split)			
Age: 5.4 years, range 0.1–18			
Source Device	Wheeze	Crackle	Normal
Yunting Model II	452	961	5159
Target Training Dataset			
ICBHI (80/20 train/test split)			
Age: 43 years, range 0.2–79.2			
Source Device $\rightarrow$ Target Device	Wheeze	Crackle	Normal
Yunting Model II $\rightarrow$ AKGC417L	452 $\rightarrow$ 392	961 $\rightarrow$ 1246	5159 $\rightarrow$ 1558
Yunting Model II $\rightarrow$ Litt3200	452 $\rightarrow$ 92	961 $\rightarrow$ 23	5159 $\rightarrow$ 260
Yunting Model II $\rightarrow$ Meditron	452 $\rightarrow$ 120	961 $\rightarrow$ 165	5159 $\rightarrow$ 826
Yunting Model II $\rightarrow$ LittC2SE	452 $\rightarrow$ 105	961 $\rightarrow$ 58	5159 $\rightarrow$ 270
Target Training Dataset			
HF_lung_v1(official train/test split)			
Age: 67.5 years, range 36.7–98.3			
Source Device $\rightarrow$ Target Device	Wheeze	Crackle	Normal
Yunting Model II $\rightarrow$ HF Type-1	452 $\rightarrow$ 3400	961 $\rightarrow$ 8258	5159 $\rightarrow$ 8289
Yunting Model II $\rightarrow$ Litt3200	452 $\rightarrow$ 3627	961 $\rightarrow$ 5536	5159 $\rightarrow$ 7312

Table 37 summarizes the training sample distribution used in the embedding alignment framework across three respiratory sound datasets. The source domain is SPRSound 2023, recorded using the Yunting Model II device and comprising pediatric subjects (mean age 5.4 years), with a class imbalance showing normal sounds over wheezes and crackles. Target domains include ICBHI and HF_Lung_v1, which differ substantially from the source in both acquisition devices and population demographics. For the ICBHI dataset, it is of note that

ICBHI official split does not cover all devices in its test set. Therefore, samples are reported in this study are under an 80/20 train–test split for multiple target stethoscopes, illustrating pronounced variability in class distributions across devices, particularly for wheeze and crackle events. Meanwhile, official split in HF_Lung_v1 further amplifies this domain shift, featuring an elderly population (mean age 67.5 years) and substantially larger target sample sizes across all classes. By reporting training samples in a source $\rightarrow$ target format, the table highlights the mismatch in data scale, device characteristics, and age distributions between source and target domains, underscoring the necessity of embedding alignment to enable robust cross-dataset and cross-device generalization.

To enable cross-domain feature alignment, each training instance in the target domain is paired with a corresponding instance from the source domain. When the number of training samples differs between the source and target datasets, the smaller dataset is randomly oversampled to match the size of the larger dataset, ensuring one-to-one correspondence during alignment. Model performance and the effectiveness of the proposed feature alignment strategy are evaluated using five-fold cross-validation, with an 80/20 train–test split applied in each fold.

Results on Source dataset

The performance of the deep learning backbone architecture has been demonstrated in our previous work [29, 35] as shown in Table 38 (Total score(%) on the blind Test set). In the IEEE BioCAS 2023 challenge [197], there are two sub-tasks of Task 1-1 and Task 1-2. While Task 1-1 aim to classify the respiratory sound events as Normal and Adventitious, Task 1-2 involves classifying these events into Normal, Rhonchi, Wheeze, Stridor, Wheeze and Crackle, Coarse Crackle, and Fine Crackle.

In this work, we shift the focus on the classification of only Wheeze, Crackle, and Normal, the results demonstrating the robust of the backbone model are presented in Table 39, when source dataset is training from Scratch. In detail, the

Table 38: Compare our deep learning backbone to the others submitted to the IEEE BioCAS 2023 challenge.

SPR 2023	Task 1-1	Task 1-2
Top 1 (Ours)	81.0	66.7
Top 2 [198]	73.3	64.6
Top 3 [199]	76.9	63.2
Top 4	72.0	59.3
Top 5	66.8	55.5
Top 6	74.8	59.9
Top 7	78.5	64.8
Top 8	75.6	46.7

backbone model exhibits a high level of performance with a sensitivity of 82.62 ± 1.1 , indicating its strong ability to correctly identify patterns in respiratory anomalies. Its specificity of 90.04 ± 0.9 demonstrates an even higher proficiency in correctly identifying normal respiratory sounds. The accuracy of the model stands at 87.94 ± 0.9 , reflecting its overall correctness in predictions. The combined AS at 86.33 ± 1.0 , HS at 86.17 ± 1.0 , and TS at 86.25 ± 1.0 highlight the model’s balanced and consistent performance in handling both anomalies and normal respiratory sounds. The low standard deviations across these metrics indicate reliable and stable detection across various evaluations.

Feature Alignment Results on Target dataset

The performance evaluation of domain adaptation methods in respiratory sounds analysis, as summarized in Table 39 (The table presents the mean $\pm$ standard error for Sensitivity, Specificity, Average Score, Total Score, and Accuracy. α_M represents different loss weight values the proposed Feature Alignment platform for each scenario. Value highlighted in bold indicate feature alignment achieved the best performance in each scenario), offers significant insights into the efficacy of feature alignment compared to direct transfer and training from scratch across different datasets. The results are categorized based on the source domain (SPR 2023 dataset) and two target domains (ICBHI 2017 and HF_Lung_v1 2022

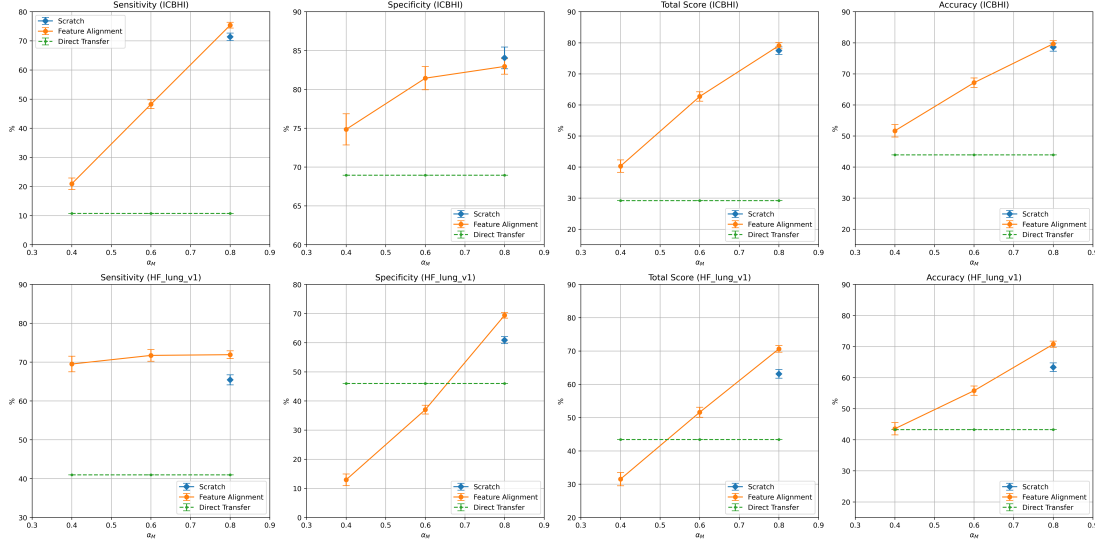


Figure 23: A visual summary of key findings on ICBHI and HF_lung_v1 dataset from Table 39.

datasets), assessing several key metrics: SE, SP, AS, HS, TS, and Acc. Table 39 also presents the impact of different loss weight values α_M on the proposed feature alignment approach.

The SPR 2023 dataset serves as the source domain, where the backbone model trained from scratch achieved a baseline accuracy of 87.94% with a sensitivity of 82.62% and specificity of 90.04%. These metrics establish a high-performing baseline, indicating that the backbone model is well-suited for classifying data within the same domain. The balanced performance across all metrics reflects the effectiveness of training from scratch when the training and testing data are derived from the same domain.

Target Domain 1: ICBHI 2017 dataset

The performance of the model significantly varies depending on the domain adaptation method employed. Direct transfer of the model without any adaptation resulted in the lowest performance, with a sensitivity of only 10.74% and an

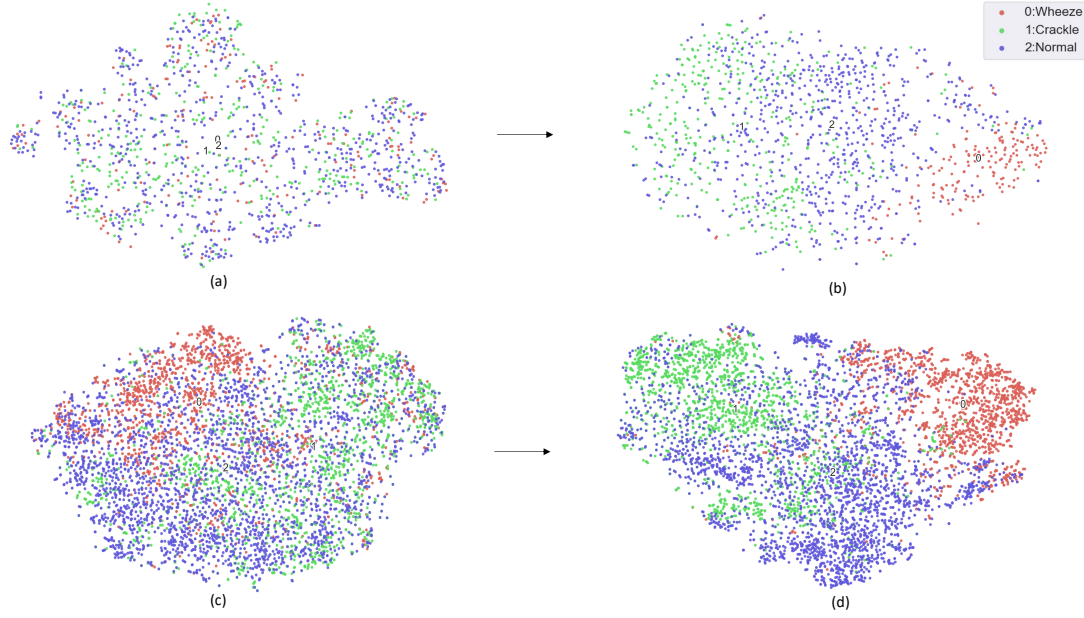


Figure 24: The *t*-SNE visualization: Embedding features extracted from Combiner block in Target Domain Stream.

accuracy of 43.93%. This substantial drop in performance highlights the importance of domain adaptation when applying a model to a different dataset with varying cross-domain differences.

In contrast, the feature alignment method demonstrated substantial improvements. At a loss weight α_M of 0.8, the model achieved the highest sensitivity (75.40%), specificity (82.96%), and accuracy (79.71%) among evaluated methods. This indicates that feature alignment is particularly effective in bridging the distribution gap between the source and target domains, allowing the model to generalize better to the ICBHI 2017 dataset. Additionally, the gradual increase in performance with higher α_M values suggests that properly tuning the loss weight enhances the alignment of feature spaces, leading to improved adaptation performance.

Target Domain 2: HF_Lung_v1 2022 Dataset

When evaluated on the HF_Lung_v1 2022 dataset, the feature alignment method

Table 39: The performance of source and target domains is evaluated using feature alignment, direct transfer, and training from scratch.

Source domain: SPR 2023 dataset							
α_M	Method	SEN	SP	AS	HS	TS	Acc
-	Scratch	82.62 ± 1.1	90.04 ± 0.9	86.33 ± 1.0	86.17 ± 1.0	86.25 ± 1.0	87.94 ± 0.9
Target domain 1: ICBHI 2017 dataset							
0.4	Feature alignment	20.94 ± 1.1	74.86 ± 1.2	47.90 ± 1.1	32.73 ± 1.5	40.31 ± 1.3	51.68 ± 0.8
0.6		48.26 ± 0.4	81.45 ± 0.5	64.86 ± 0.6	60.61 ± 0.3	62.73 ± 0.3	67.18 ± 0.4
0.8		75.40 ± 0.2	82.96 ± 0.3	79.18 ± 0.4	79.00 ± 0.4	79.19 ± 0.1	79.71 ± 0.2
-	Scratch	71.40 ± 0.3	84.06 ± 0.4	77.73 ± 0.3	77.21 ± 0.3	77.47 ± 0.4	77.62 ± 0.3
-	Direct transfer	10.74 ± 0.3	68.95 ± 0.4	39.85 ± 0.3	18.59 ± 0.3	29.22 ± 0.4	43.93 ± 0.3
Target domain 2: HF_Lung_v1 2022 dataset							
0.4	Feature alignment	69.52 ± 0.3	12.95 ± 0.4	41.24 ± 0.3	21.83 ± 0.3	31.53 ± 0.4	43.57 ± 0.3
0.6		71.74 ± 0.4	37.04 ± 0.5	54.39 ± 0.4	48.86 ± 0.3	51.62 ± 0.5	55.82 ± 0.4
0.8		71.93 ± 0.4	69.43 ± 1.2	70.68 ± 0.5	70.65 ± 0.5	70.66 ± 0.5	70.78 ± 1.2
-	Scratch	65.45 ± 0.3	60.91 ± 0.4	63.18 ± 0.3	63.10 ± 0.3	63.14 ± 0.4	63.37 ± 0.3
-	Direct transfer	40.99 ± 0.3	45.99 ± 0.4	43.49 ± 0.3	43.35 ± 0.3	43.42 ± 0.4	43.28 ± 0.3

again outperformed the other approaches. For example, training from scratch yielded an accuracy of 63.37%, which was higher than that achieved with direct transfer (43.28%) but lower than the best performance obtained through feature alignment.

With α_M of 0.8, the feature alignment method resulted in a sensitivity of 71.93% and specificity of 69.43%, achieving an overall accuracy of 70.78%. These results, while still showing an improvement over direct transfer, indicate a more challenging domain adaptation scenario compared to the ICBHI 2017 dataset. The reduced sensitivity and specificity suggest that the HF_Lung_v1 2022 dataset may have more complex or diverse features that are not as easily aligned with those of the source domain. Nevertheless, the feature alignment method outperforms the approach of training model from scratch. It indicates an improvement in cross-domain prediction accuracy.

As illustrated in Figure 23 (highlights the performance in SE, SP, TS and Acc between feature alignment, and training from scratch, compared to direct transfer. The results are depicted with experiments on different value of α_M ranging from

0.4 to 0.8.), the results clearly demonstrate the superiority of the feature alignment method in enhancing the generalization of models across different datasets, particularly when there are notable differences between the source and target domains. The consistent improvement in accuracy, sensitivity, and specificity with increasing α_M values further validates the importance of the loss weight to optimize feature alignment. The relatively poor performance of direct transfer across both target domains highlights the limitations of applying models trained on one dataset to another without adaptation. This shows the need for domain adaptation techniques such as our proposed feature alignment, which can mitigate the distribution discrepancies that often exist between different datasets.

Table 40 demonstrates the effectiveness of embedding alignment across different recording devices and target datasets. On the ICBHI dataset, increasing the alignment weight from $\alpha_M = 0.4$ to $\alpha_M = 0.8$ yields substantial accuracy improvements across all devices, with gains of more than 30% for AKGC417L (46.81% $\rightarrow$ 77.37%), 23% for Litt3200 (63.72% $\rightarrow$ 87.25%), 20% for Meditron (64.35% $\rightarrow$ 84.77%), and a significant improvement of 37% for LittC2SE (47.00% $\rightarrow$ 84.61%).

Notably, sensitivity increases substantially from 35.80% to 73.82% despite the markedly reduced number of Crackle samples available in the target dataset compared with the source domain. This pronounced improvement indicates that the proposed feature alignment framework is capable of effectively transferring discriminative representations across domains, even under conditions of significant class imbalance and limited target-domain supervision.

On the HF_Lung_v1 dataset, although absolute performance is lower, consistent gains are still observed with stronger alignment. For HF Type-1 device, accuracy and specificity improve nearly 10%, while Litt3200 shows an accuracy increase of more than 5% and balanced improvements across AS, HS, and TS metrics from $\alpha_M = 0.4$ to $\alpha_M = 0.8$.

Across both target datasets, figures in AS, HS and TS increase by approximately 10% to 35% under stronger alignment factor (e.g. from $\alpha_M = 0.4$ to $\alpha_M = 0.8$) for HF_Lung_v1 dataset and ICBHI dataset, respectively. This indicates that stronger embedding alignment effectively reduces cross-device and cross-dataset domain discrepancies, leading to more balanced classification behavior across sensitivity and specificity. Notably, the most pronounced relative improvements are observed for devices associated with greater initial domain mismatch, suggesting that the proposed framework is particularly beneficial in challenging transfer scenario under conditions of device imbalance and limited target-domain supervision, and different recording environments.

In Figure 24, it is clear that the class-wise distributions are aligned to improve cross-domain detection. In detail, (a) From ICBHI dataset after Direct Transfer from the Source dataset, (b) From ICBHI dataset after Feature Alignment from the Source dataset, (c) From HF_lung_v1 dataset after Direct Transfer from the Source dataset, (d) From HF_lung_v1 dataset after Feature Alignment from the Source dataset. For instance, we visualize the embedding spaces by the T-distributed Stochastic Neighbor Embedding (t-SNE) plot of Combined-features extracted from Combiner block in Target Domain Stream in both cases: Direct transfer and Embedding alignment. In ICBHI dataset, Figure 24 (a) ICBHI dataset after Direct Transfer from the Source dataset. Figure 24 (b) ICBHI dataset after Embedding alignment from the Source dataset. In HF_Lung_v1 dataset, Figure 24 (c) HF_Lung_v1 dataset after Direct Transfer from the Source dataset, Figure 24 (d) HF_Lung_v1 dataset after Embedding Alignment from the Source dataset.

We can clearly see the improvement in each Target dataset when Feature Alignment method is applied. This can be explained by two reasons. Firstly, within class-wise approach, when all pairs of samples from both Source and Target dataset are trained, we can observe that Combined-features extracted from

Combiner block within a specific region of the feature space provide more precise location for alignment for single class such as Wheeze, Crackle, or Normal. Secondly, the feature alignment technique enables the preservation of knowledge acquired from the source domain. It aligns target domain features with corresponding source domain features, directly contributes to the efficiency of adaptation.

Furthermore, the comparison between the ICBHI 2017 and HF_Lung_v1 2022 datasets reveals that the effectiveness of domain adaptation varied depending on the nature and complexity of the target domain. While feature alignment significantly improved performance on the ICBHI 2017 dataset, the enhancements were more limited on the HF_Lung_v1 2022 dataset, where labeling quality has been acknowledged as a persisting challenge in [57]. This discrepancy suggests that factors beyond domain alignment, such as annotation consistency and dataset-specific characteristics, may influence adaptation outcomes. Therefore, future research should explore more sophisticated or dataset-specific adaptation techniques, potentially incorporating improved labeling strategies, advanced feature extraction methods, or hybrid approaches to further enhance performance across diverse datasets.

Table 40: Feature alignment performance by recording devices on target datasets (ICBHI and HF_Lung_v1).

Dataset	Device	Metric	α_M		
			0.4	0.6	0.8
ICBHI	AKGC417L	Acc	46.81±1.4	67.10±1.2	77.37±1.2
		SEN	35.80±2.7	72.34±1.8	73.82±1.7
		SPEC	59.06±1.3	61.26±2.1	81.31±1.6
		AS	47.43±2.5	66.80±1.1	77.57±2.2
		HS	44.58±3.8	66.34±2.1	77.39±1.8
		TS	46.01±1.9	66.57±2.2	77.48±1.5
	Litt3200	Acc	63.72±2.2	65.68±2.3	87.25±1.3
		SEN	23.07±2.4	76.92±3.5	65.38±0.7
		SPEC	77.63±3.1	61.84±2.9	94.73±2.1
		AS	50.35±2.8	69.38±3.1	80.06±1.5
		HS	35.57±3.6	68.56±3.4	77.37±3.1
		TS	42.96±2.9	68.97±2.6	78.72±2.1
	Meditron	Acc	64.35±2.6	79.93±2.1	84.77±1.5
		SEN	14.10±1.9	70.51±1.7	67.94±0.8
		SPEC	82.93±2.5	83.41±2.7	90.99±0.8
		AS	48.52±2.1	76.96±2.6	79.47±1.4
		HS	24.10±1.9	76.42±2.2	77.80±1.9
		TS	36.31±2.2	76.69±2.5	78.64±2.6
	LittC2SE	Acc	47.00±1.9	70.08±2.4	84.61±1.5
		SEN	55.00±2.9	42.50±2.8	72.50±0.8
		SPEC	42.85±2.2	84.41±2.1	90.90±0.8
		AS	48.92±2.1	63.45±3.2	81.70±1.4
		HS	48.17±2.5	56.53±2.1	80.66±1.9
		TS	48.54±2.6	59.99±2.4	81.18±2.6
HF_Lung_v1	HF Type-1	Acc	44.52±1.2	50.56±1.3	54.34±1.8
		SEN	43.26±1.8	56.13±2.1	43.00±1.3
		SPEC	46.06±0.9	43.75±1.9	68.20±1.5
		AS	44.66±1.3	49.94±1.2	55.60±1.7
		HS	44.61±1.6	49.17±1.5	52.74±1.6
		TS	44.63±1.3	49.55±1.4	54.17±1.5
	Litt3200	Acc	71.25±2.3	73.29±1.5	76.96±1.3
		SEN	88.39±3.1	72.17±1.7	81.99±1.7
		SPEC	51.42±2.1	74.59±1.9	71.14±1.5
		AS	69.90±2.8	73.38±1.7	76.56±1.4
		HS	65.01±2.2	73.36±1.5	76.18±1.5
		TS	67.45±2.6	73.37±1.6	76.37±1.4

Chapter 6

Conclusion and Future Work

6.1 Overview

This thesis presented a comprehensive study on the classification of respiratory anomalies using advanced audio feature extraction and deep learning techniques applied to CRAA. By leveraging multiple spectrogram and scalogram representations and various deep learning architectures—including joint learning models, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks demonstrated that deep learning can be effectively used to detect and classify abnormal respiratory sounds with high accuracy and robustness. Across the developed models, emphasis was placed on enhancing feature extraction from respiratory audio signals, improving classification performance through a variety of spectrograms, and evaluating generalization in real-world respiratory conditions, including those related to COVID-19. Each proposed method contributed to addressing the inherent challenges in respiratory sound analysis, such as variability in respiratory audio quality, class imbalance, and the need for cross-domain classification.

The exploration starts by introducing and systematically evaluating various

framework including respiratory audio feature extraction and back-end deep learning architecture in CRAA. This initial phase of introducing Audio Feature Extraction in Chapter 3 is crucial as it assesses how different techniques can enhance each other’s strengths and mitigate weaknesses when combined. In details, we initiated our research in Section 3.1 and 3.2 with comprehensive acoustic signal processing experiments using spectrograms, focusing on the acoustic scene classification. The evaluation of diverse acoustic modeling techniques in this preliminary phase establishes a robust foundation for CRAA. Subsequently, the research transitions to providing detailed insights into respiratory audio feature extraction, which is critical for the analysis of advanced deep learning architectures. This understanding of complementarity is pivotal, as it lays the foundational groundwork for developing more sophisticated ensemble systems in Chapter 4.

Following this initial step, a combined system using C-DNN and Autoencoder networks in Section 4.1 has been evaluated to pinpoint the potential of using deep learning and Gammatone spectrogram to classify the respiratory anomaly cycles. Inspired by this success, Section 4.2 examines the role of ensemble methods in CRAA. This exploration highlights how different spectrogram using Log-Mel and Wavelet-based spectrograms can influence the performance in CRAA systems, potentially leading to improvements in accuracy and reliability. This novel exploration highlights how an ensemble of inception-based and transfer-learning-based deep learning frameworks, outperforms the state-of-the-art systems, then validate this application of deep learning for early detecting of respiratory anomaly. However, instead of simply combining the spectrogram results through concatenation, Section 4.3 proposes a Linear Combination. This new approach has the distinct advantage of being able to control the influence of each individual spectrogram equally during training. To build on this, Section 4.4 introduced an Inception-residual network enhanced with spatio-temporal focusing and multi-head attention for identifying respiratory anomalies. This model achieved Top-1 performance in

the IEEE BioCAS 2023 challenge, demonstrating a key insight: the higher number of time frames are more effective in the Wavelet (Bump) spectrogram for event-level tasks, and in the Wavelet (Morse) spectrogram for recording-level tasks. In conclusion, the model proves its ability to learn temporal information effectively when the time frame of the spectrograms is extended.

Building upon the solid foundation established in Chapter 4, this thesis continues in Chapter 5 by introducing a novel cross-domain adaptation for CRAA. Section 5.1 introduces an exploration of how to effectively extract well-represented features for breathing, cough, and speech sound inputs using pre-trained models. Through extensive experimentation, we were able to achieve a robust framework for detecting COVID-19 that surpasses the performance of state-of-the-art systems. Additionally, this thesis continues in Section 5.2 by introducing a novel domain adaptation framework designed to address the significant challenges of subjectivity and variability in respiratory auscultation. Our approach features a linear combination method for scalograms and a unique embedding-based alignment method, which significantly enhances cross-domain diagnostic accuracy. This framework achieved strong and competitive results on both pediatric and adult data, proving its ability to robustly generalize to new datasets and effectively transfer knowledge across disparate domains. This is a crucial capability for reliable remote respiratory care.

While this thesis does not claim to definitively resolve all long-standing challenges in CRAA, it offers significant insights and methodologies to enhance and improve current approaches. The introduced systems and frameworks are specifically designed to tackle practical problems like clinical artifacts, environmental noise, audio quality variation, and adaptability, thereby advancing the field of CRAA. Essentially, the results validate the potential of audio-based deep learning systems as non-invasive, cost-effective, and scalable tools for supporting early diagnosis and continuous monitoring of respiratory conditions. These findings pave

the way for integrating such systems into telehealth platforms and mobile health applications, particularly in low-resource settings.

6.2 Summary of Contributions and Significance

The key contributions of this thesis are summarized as follows:

1. Design and evaluation of multiple deep learning architectures for respiratory sound analysis. This thesis introduces and evaluates a suite of deep learning models tailored to classify respiratory sounds. These models include joint-learning architectures, ensemble frameworks, spatio-temporal attention mechanisms, and Inception-Residual networks with specific challenges in respiratory anomaly detection from audio signals. (Outcome from RO1, RO2 and RO5)
2. Present the utilization of multi-spectrogram features for comprehensive audio representation. The thesis advocates a multi-spectrogram approach by combining representations such as Log-mel, STFT, MFCC, Gammatone and CQT spectrograms. This strategy demonstrate the use of multi-spectrogram can enhances the diversity and richness information of time-frequency features. (Outcome from RO1)
3. Introduce the development of a multi-objective loss function. A novel balanced loss function was introduced to simultaneously optimize multiple learning objectives, including classification accuracy, combined feature embeddings, and contrastive learning separability. This optimization led to more stable training and better generalization. (Outcome from RO3)
4. Implement an ensemble learning techniques to enhance predictive reliability. In details, an ensemble of diverse deep learning models was constructed to aggregate their probability outputs. This approach reduces predictive

variance and improves overall robustness for respiratory sound classification. (Outcome from RO2)

5. Propose a spatio-temporal attention mechanism for targeted feature focus. A spatio-temporal attention module was integrated into the model pipeline, enabling dynamic focus on the most informative regions within spectrograms. This not only improved classification accuracy but also enhanced interpretability. (Outcome from RO1 and RO2)
6. Implement a wide range of audio-based deep learning frameworks applied to the breathing, cough, and speech sounds for detecting COVID-19. Conducting extensive experiments and pinpoint the most effective approach related to the extraction of well-represented features for each type of breathing, cough, speech sound input. The framework is reliable and robust for detecting COVID-19 which outperforms the state-of-the-art systems and is potential for a real-life application. (Outcome from RO1 and RO2)
7. Present an embedding alignment framework for discriminative representation learning to refine latent feature representations. This method encourages more discriminative embeddings, improving the separability across varied recording conditions by clinical environments and patient demographics. (Outcome from RO5)

6.3 Impact on Respiratory Disease Detection

The research presented in this thesis has the potential to make a significant impact on the field of respiratory disease detection, particularly by introducing non-invasive, scalable, and intelligent solutions based on deep learning and audio analysis.

1. Advancing Non-Invasive Diagnostic Tools

By focusing on the analysis of respiratory sounds, this work contributes to the development of diagnostic systems that do not rely on expensive or invasive procedures such as imaging or laboratory tests. This opens new avenues for early screening and continuous monitoring of respiratory conditions in both clinical and remote settings.

2. Supporting Early Detection and Monitoring

The proposed deep learning models have demonstrated strong performance in detecting various respiratory anomalies, including those related to infectious diseases such as Asthma, COPD and COVID-19. These tools can facilitate early identification of abnormal respiratory patterns, potentially improving outcomes through earlier intervention.

3. Enabling Scalable and Accessible Healthcare Solutions

The deployment of light-weight models allows integration into mobile devices or telemedicine platforms, offering practical solutions for regions with limited access to healthcare infrastructure. This is consistent with the global initiative to extend CRAA applications across broader healthcare systems.

4. Enhancing Diagnostic Accuracy and Consistency

Traditional auscultation is highly dependent on the clinician's expertise and experience. By leveraging automated deep learning approaches, this research introduces a CRAA framework that can complement clinical decision-making and reduce diagnostic variability.

5. Laying the Groundwork for Future Clinical Applications

The methods proposed in this thesis serve as a foundation for the future development of intelligent diagnostic assistants capable of analyzing respiratory sounds in real time. Such systems may be integrated into smart stethoscopes, wearable devices, or hospital information systems, thereby extending their impact across multiple layers of healthcare delivery.

In summary, the outcomes of this research contribute not only to the academic understanding of audio-based classification models but also to the practical advancement of accessible and reliable technologies for respiratory disease detection.

6.4 Future Directions

While the research presented in this thesis demonstrates promising results for the classification of respiratory anomalies using deep learning, several directions remain for further investigation and improvement. As an immediate extension of the current work, enhancing the interpretability of the proposed models should be prioritized. Improving model transparency and explainability is essential for increasing clinical trust and facilitating the integration of automated respiratory sound analysis into healthcare settings. In the longer term, future research should focus on improving the generalization capability of the developed models across diverse datasets, recording devices, and acoustic environments. Addressing these challenges would enable more robust performance in real-world clinical and home-based scenarios and support the development of scalable respiratory monitoring systems.

1. Interpretability

Future research should focus on advanced integration strategies for the multi-spectrogram features developed in this work. While Chapter 3 establishes the utility of diverse representations (Log-Mel, STFT, MFCC, CQT, Gammatone, Wavelet-based), the next logical step is to explore non-linear and adaptive fusion techniques. Rather than learning feature from defined respiratory audio representation as mentioned in Section 4.1, 4.2, 4.3 and 4.4, future models could incorporate attention-based mechanisms and learnable audio feature extraction within the framework proposed in Chapter 4.

This can allow to dynamically prioritize the most informative time-frequency features based on the specific respiratory sound event and show the interpretability for CRAA.

2. Generalization

Future research should concentrate on extending the generalization capabilities of the Chapter 5 cross-domain adaptation framework. A critical next step involves investigating unsupervised or semi-supervised domain adaptation techniques to mitigate the reliance on labeled target-domain data, which is often limited in real-world clinical settings. This could include exploring advanced Generative Adversarial Networks (GANs) or self-supervised learning methods to enhance the embedding-based alignment and feature disentanglement for the cough, speech, and breathing sounds. Furthermore, to fully address the variability challenges, research should focus on developing a meta-learning framework that can consider not only respiratory audio but also demographic information (e.g., age, gender, geographic location) as a critical auxiliary input. Integrating these multi-modal inputs will allow the model to learn a more robust, conditional representation, thereby facilitating faster and more nuanced adaptation to new and diverse patient populations and clinical environments.

6.5 Closing Remarks

This thesis has explored the integration of deep learning techniques with respiratory sound analysis to develop accurate and scalable systems for detecting respiratory anomalies. The work has demonstrated that with thoughtful model design, appropriate audio feature extraction, and careful evaluation, it is possible to build intelligent systems capable of supporting respiratory health assessment in both clinical and home-based settings. Beyond the technical contributions,

this research reflects the broader potential of AI-based platforms to transform traditional healthcare practices. As digital health tools become increasingly vital, especially in the context of global health challenges such as pandemics and limited medical access, innovations in audio-based diagnostics can offer timely, non-invasive, and cost-effective solutions.

However, the journey does not end here; there are still challenges to be addressed, and the progress made in this thesis lays a solid foundation for future developments in CRAA for respiratory monitoring. It has the potential to provide the insight and methodologies to inspire further interdisciplinary research and contribute meaningfully to the advancement of accessible and equitable healthcare technologies.

Appendix A

Mathematical definitions

This appendix provides details on the computation of network layers used in this thesis such as convolutional (Cv), batch normalization (BN), rectify linear unit (ReLU), dropout (Dr), fully connected (FC), and Softmax layers. Note that “ $\times$ ” is used for matrix product.

A.1 Computation of Network Layers

if $\mathbf{X}[\mathbf{F}, \mathbf{T}, \mathbf{C}] \in \mathbb{R}^{\mathbb{F} \times \mathbb{T} \times \mathbb{C}}$ and $\mathbf{x} \in \mathbb{R}^N$ are considered as an input tensor and input vector, where F, T, C are frequency, time, channel dimensions of the input tensor and N is the dimension of input vector, the computation of network layers used in thesis is described in detail as below,

A.1.1 Rectify Linear Unit (ReLU) Layer

Relu function takes each pixel $x_{f,t,c}$ of input tensor $\mathbf{X}$ and thus returns the output $f_{Relu}(x_{f,t,c})$ as,

$$f_{ReLU}(x_{f,t,c}) = \begin{cases} x_{f,t,c}, & \text{if } x_{f,t,c} > 0 \\ 0, & \text{if } x_{f,t,c} \leq 0 \end{cases} \quad (60)$$

As regards input vector $\mathbf{x}[x_1, \dots, x_N]$, this function takes x_n and returns

$$f_{ReLU}(x_n) = \begin{cases} x_n, & \text{if } x_n > 0 \\ 0, & \text{if } x_n \leq 0 \end{cases} \quad (61)$$

A.1.2 Dropout Layer

Dropout function f_{Drop} for input tensor $\mathbf{X}$ is computed by,

$$f_{Drop}(\mathbf{X}) = \mathbf{X}\mathbf{D} \quad (62)$$

and for input vector $\mathbf{x}$ as

$$f_{Drop}(\mathbf{x}) = \mathbf{x}\mathbf{d} \quad (63)$$

where $\mathbf{D} \in \mathbb{R}^{\mathbb{F} \times \mathbb{T} \times \mathbb{C}}$ is a matrix with similar dimension of input tensor $\mathbf{X}$ and $\mathbf{d} \in \mathbb{R}^{\mathbb{N}}$ is a vector with similar dimension of input vector $\mathbf{x}$. $\mathbf{D}$ and $\mathbf{d}$ are generated by $\mathbf{D}/\mathbf{d} \sim \text{Bernoulli}(1-p)$ where p is the percentage of input dropped.

A.1.3 Batch Normalization Layer

If a batch of B tensor is described by $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_B\}$, where $\mathbf{X}_b$ is a tensor input, the batch normalization function takes a tensor $\mathbf{X}_b$, and thus returns an output $f_{Batch}(\mathbf{X}_b)$ as

$$f_{Batch}(\mathbf{X}_b) = \lambda \frac{\mathbf{X}_b - \mu}{\sqrt{\sigma + \epsilon}} \quad (64)$$

where μ and σ are defined by

$$\mu = \frac{1}{B} \sum_{b=1}^B \mathbf{X}_b \quad (65)$$

$$\sigma = \frac{1}{B} \sum_{b=1}^B (\mathbf{X}_b - \mu)^2 \quad (66)$$

and λ and ϵ are scale and ship parameters that are learned during training process. In this thesis, the batch normalization function is only applied over input tensors and across the channel dimension.

A.1.4 Fully Connected Layer

In this thesis, the fully-connected layer is applied for input vector. The output vector $\mathbf{y}[y_1, y_2, \dots, y_M]$ with output dimension of M is computed by

$$y_m = \sum_{n=0}^N w_{m,n} x_n + b_m \quad (67)$$

where pixel $w_{m,n}$ in the coefficient matrix $\mathbf{W}[\mathbf{M}, \mathbf{N}]$ and b_m in bias vector $\mathbf{b}[b_1, \dots, b_M]$ are trainable parameters.

A.1.5 Softmax Layer

In this thesis, the Softmax layer is applied on input vectors. The output vector $\mathbf{y} = [y_1, y_2, \dots, y_N]$ of this layer is computed by

$$y_n = \frac{\exp(x_n)}{\sum_{n=1}^N \exp(x_n)} \quad (68)$$

Note that the output and input vectors of Softmax layer have same dimension of N .

A.1.6 Convolutional Layer

In this thesis, convolutional layer is applied on input tensors. if C' kernels with size of $[K, P]$ are applied on the input tensor $\mathbf{X}[\mathbf{F}, \mathbf{T}, \mathbf{C}]$ in a convolutional layer, the output tensor has size of $\mathbf{Y}[\mathbf{F}, \mathbf{T}, \mathbf{C}']$ (note that the frequency F and time T

dimensions are remained by adding zero padding), a pixel $y_{f,t,c'}$ of output tensor $\mathbf{Y}$ is computed by,

$$y_{f,t,c'} = \sum_{k=1}^K \sum_{p=1}^P \sum_{c=1}^C w_{k,p,c,c'} x_{k,p,c} + b_{c'} \quad (69)$$

where pixel $x_{k,p,c}$ is in tensor input $\mathbf{X}[\mathbf{F}, \mathbf{T}, \mathbf{C}]$; pixel $w_{k,p,c,c'}$ in coefficient matrix $\mathbf{W}[\mathbf{K}, \mathbf{P}, \mathbf{C}, \mathbf{C}']$ and $b_{c'}$ in bias vector $\mathbf{b} = [b_1, \dots, b_{C'}]$ are trainable parameters.

A.1.7 Multi-head Attention Layer

In this thesis, given query, key, and value matrices $Q \in \mathbb{R}^{n \times d_q}$, $K \in \mathbb{R}^{m \times d_k}$, and $V \in \mathbb{R}^{m \times d_v}$, the scaled dot-product attention is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^\top}{\sqrt{d_k}} \right) V,$$

For multi-head attention with h heads, the input is linearly projected h times to lower-dimensional subspaces:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \quad \text{for } i = 1, \dots, h,$$

where $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ are learned projection matrices.

The outputs of all heads are concatenated and projected again:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O,$$

where $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$ is the output projection matrix.

Bibliography

- [1] BritishLungFoundation, “Lung disease in the uk,” 2023.
- [2] H. Pasterkamp, S. S. Kraman, and G. R. Wodicka, “Respiratory sounds: advances beyond the stethoscope,” *American journal of respiratory and critical care medicine*, vol. 156, no. 3, pp. 974–987, 1997.
- [3] E. Andrès, R. Gass, A. Charloux, C. Brandt, and A. Hentzler, “Respiratory sound analysis in the era of evidence-based medicine and the world of medicine 2.0,” *Journal of medicine and life*, vol. 11, no. 2, p. 89, 2018.
- [4] A. Sovijarvi, F. Dalmasso, J. Vanderschoot, L. Malmberg, G. Righini, and S. Stoneman, “Definition of terms for applications of respiratory sounds,” *European Respiratory Review*, vol. 10, no. 77, pp. 597–610, 2000.
- [5] M. Mussell, “The need for standards in recording and analysing respiratory sounds,” *Medical and Biological Engineering and Computing*, vol. 30, no. 2, pp. 129–139, 1992.
- [6] V. Suttels, S. G. Da Costa, E. Garcia, T. Brahier, M.-A. Hartley, G. Agodokpessi, P. Wachinou, F. Fasseur, and N. Boillat-Blanco, “Barriers and facilitators to implementation of point-of-care lung ultrasonography in a tertiary centre in benin: a qualitative study among general physicians and pneumologists,” *BMJ open*, vol. 13, no. 6, p. e070765, 2023.
- [7] H. Hafke-Dys, A. Breborowicz, P. Kleka, J. Kociński, and A. Biniakowski, “The accuracy of lung auscultation in the practice of physicians and medical students,” *PLoS One*, vol. 14, no. 8, p. e0220606, 2019.
- [8] H. Melbye, L. Garcia-Marcos, P. Brand, M. Everard, K. Priftis, and H. Pasterkamp, “Wheezes, crackles and rhonchi: simplifying description of lung sounds increases the agreement on their classification: a study of 12

- physicians' classification of lung sounds from video recordings," *BMJ open respiratory research*, vol. 3, no. 1, p. e000136, 2016.
- [9] S. Mangione and L. Z. Nieman, "Pulmonary auscultatory skills during training in internal medicine and family practice," *American journal of respiratory and critical care medicine*, vol. 159, no. 4, pp. 1119–1124, 1999.
- [10] M. Mehmood, H. L. Abu Grara, J. S. Stewart, and F. A. Khasawneh, "Comparing the auscultatory accuracy of health care professionals using three different brands of stethoscopes on a simulator," *Medical Devices: Evidence and Research*, pp. 273–281, 2014.
- [11] J. J. Seah, J. Zhao, D. Y. Wang, and H. P. Lee, "Review on the advancements of stethoscope types in chest auscultation," *Diagnostics*, vol. 13, no. 9, p. 1545, 2023.
- [12] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone, "A neural network-based method for respiratory sound analysis and lung disease detection," *Applied Sciences*, vol. 12, no. 8, p. 3877, 2022.
- [13] J. F. Kanga and S. S. Kraman, "Comparison of the lung sound frequency spectra of infants and adults," *Pediatric pulmonology*, vol. 2, no. 5, pp. 292–295, 1986.
- [14] H. Pasterkamp, R. E. Powell, and I. Sanchez, "Lung sound spectra at standardized air flow in normal infants, children, and adults.," *American journal of respiratory and critical care medicine*, vol. 154, no. 2, pp. 424–430, 1996.
- [15] R. Palaniappan, K. Sundaraj, and N. U. Ahamed, "Machine learning in lung sound analysis: a systematic review," *Biocybernetics and Biomedical Engineering*, vol. 33, no. 3, pp. 129–135, 2013.

- [16] J. Zhang, W. Ser, J. Yu, and T. Zhang, "A novel wheeze detection method for wearable monitoring systems," in *2009 International Symposium on Intelligent Ubiquitous Computing and Education*, pp. 331–334, IEEE, 2009.
- [17] S.-Y. Jung, C.-H. Liao, Y.-S. Wu, S.-M. Yuan, and C.-T. Sun, "Efficiently classifying lung sounds through depthwise separable cnn models with fused stft and mfcc features," *Diagnostics*, vol. 11, no. 4, p. 732, 2021.
- [18] G. Chambres and et al., "Automatic detection of patient with respiratory diseases using lung sound analysis," in *Proc. CBMI*, pp. 1–6, 2018.
- [19] T. Okubo *et al.*, "Classification of healthy subjects and patients with pulmonary emphysema using continuous respiratory sounds," in *Proc. EMBC*, pp. 70–73, 2014.
- [20] P. Lubaib and K. A. Muneer, "The heart defect analysis based on pcg signals using pattern recognition techniques," *Procedia Technology*, vol. 24, pp. 1024–1031, 2016.
- [21] M. Grønnesby and et al., "Feature extraction for machine learning based crackle detection in lung sounds from a health survey," *preprint arXiv:1706.00005*, 2017.
- [22] A. Moukadem, A. Dieterlen, N. Hueber, and C. Brandt, "A robust heart sounds segmentation module based on s-transform," *Biomedical Signal Processing and Control*, vol. 8, no. 3, pp. 273–281, 2013.
- [23] S. Datta, A. D. Choudhury, P. Deshpande, S. Bhattacharya, and A. Pal, "Automated lung sound analysis for detecting pulmonary abnormalities," in *2017 39th Annual International Conference of the Ieee Engineering in Medicine and Biology Society (Embc)*, pp. 4594–4598, IEEE, 2017.

- [24] M. T. García-Ordás, J. A. Benítez-Andrades, I. García-Rodríguez, C. Benavides, and H. Alaiz-Moretón, “Detecting respiratory pathologies using convolutional neural networks and variational autoencoders for unbalancing data,” *Sensors*, vol. 20, no. 4, p. 1214, 2020.
- [25] D. Ngo, L. Pham, T. Hoang, Kolozali, and D. Jarchi, “Audio-based deep learning frameworks for detecting covid-19,” in *2022 30th European Signal Processing Conference (EUSIPCO)*, pp. 1233–1237, 2022.
- [26] Y. Kim, Y. Hyon, S. S. Jung, S. Lee, G. Yoo, C. Chung, and T. Ha, “Respiratory sound classification for crackles, wheezes, and rhonchi in the clinical field using deep learning,” *Scientific reports*, vol. 11, no. 1, p. 17186, 2021.
- [27] L. Pham, D. Ngo, T. Hoang, A. Schindler, and I. McLoughlin, “An ensemble of deep learning frameworks applied for predicting respiratory anomalies,” *arXiv preprint arXiv:2201.03054*, 2022.
- [28] Z. Neili and K. Sundaraj, “A comparative study of the spectrogram, scalogram, melspectrogram and gammatonegram time-frequency representations for the classification of lung sounds using the icbhi database based on cnns,” *Biomedical Engineering/Biomedizinische Technik*, vol. 67, no. 5, pp. 367–390, 2022.
- [29] D. Ngo, L. Pham, H. Phan, M. Tran, and D. Jarchi, “A deep learning architecture with spatio-temporal focusing for detecting respiratory anomalies,” in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–5, IEEE, 2023.
- [30] J. Acharya and A. Basu, “Deep neural network for respiratory sound classification in wearable devices enabled by patient specific model tuning,” *IEEE transactions on biomedical circuits and systems*, vol. 14, no. 3, pp. 535–544, 2020.

- [31] L. Pham and et al., “Cnn-moe based framework for classification of respiratory anomalies and lung disease detection,” *IEEE journal of biomedical and health informatics*, vol. 25, no. 8, pp. 2938–2947, 2021.
- [32] D. Ngo, L. Pham, A. Nguyen, B. Phan, K. Tran, and T. Nguyen, “Deep learning framework applied for predicting anomaly of respiratory sounds,” in *2021 International Symposium on Electrical and Electronics Engineering (ISEE)*, pp. 42–47, IEEE, 2021.
- [33] K. Minami, , and et al., “Automatic classification of large-scale respiratory sound dataset based on convolutional neural network,” in *Proc. ICCASP*, pp. 804–807, 2019.
- [34] D. Perna and et al., “Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks,” in *Proc. CBMS*, pp. 50–55, 2019.
- [35] D. Ngo, L. Pham, H. Phan, M. Tran, D. Jarchi, and Kolozali, “An inception-residual-based architecture with multi-objective loss for detecting respiratory anomalies,” in *2023 IEEE 25th International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6, 2023.
- [36] S. B. Shuvo, S. N. Ali, S. I. Swapnil, T. Hasan, and M. I. H. Bhuiyan, “A lightweight cnn model for detecting respiratory diseases from lung auscultation sounds using emd-cwt-based hybrid scalogram,” *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2595–2603, 2020.
- [37] T. Nguyen and F. Pernkopf, “Lung sound classification using co-tuning and stochastic normalization,” *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 9, pp. 2872–2882, 2022.

- [38] H. Guan and M. Liu, “Domain adaptation for medical image analysis: a survey,” *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 3, pp. 1173–1185, 2021.
- [39] P. Singhal, R. Walambe, S. Ramanna, and K. Kotecha, “Domain adaptation: challenges, methods, datasets, and applications,” *IEEE access*, vol. 11, pp. 6973–7020, 2023.
- [40] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, “Universal domain adaptation in fault diagnostics with hybrid weighted deep adversarial learning,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 7957–7967, 2021.
- [41] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, “Supervised contrastive learning,” *Advances in neural information processing systems*, vol. 33, pp. 18661–18673, 2020.
- [42] J. Li, G. Li, Y. Shi, and Y. Yu, “Cross-domain adaptive clustering for semi-supervised domain adaptation,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2505–2514, 2021.
- [43] D. Ngo, Y. Raykov, M. Shaban, D. Jarchi, A. Gratiot, and C. Sheppard, “Advancements in early airway disease detection using vowel phonation with smartphone,” 2024.
- [44] D. Ngo, H. Hoang, A. Nguyen, T. Ly, and L. Pham, “Sound context classification basing on join learning model and multi-spectrogram features,” *International Journal of Computing*, vol. 21, no. 2, pp. 58–270, 2022.
- [45] L. Pham, D. Ngo, K. Tran, T. Hoang, A. Schindler, and I. McLoughlin, “An ensemble of deep learning frameworks for predicting respiratory anomalies,” in *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 4595–4598, IEEE, 2022.

- [46] M. Deng, T. Meng, J. Cao, S. Wang, J. Zhang, and H. Fan, “Heart sound classification based on improved mfcc features and convolutional recurrent neural networks,” *Neural Networks*, vol. 130, pp. 22–32, 2020.
- [47] M. Akay, Y. M. Akay, G. Landesberg, W. Welkowitz, and D. Sapoznikov, “Time-frequency analysis of heart rate fluctuations during carotid surgery using the short term fourier and fast wavelet transforms,” *BME*, vol. 11, no. 1, pp. 24–33, 1997.
- [48] A. I. Taloba and R. Matoog, “Detecting respiratory diseases using machine learning-based pattern recognition on spirometry data,” *Alexandria Engineering Journal*, vol. 113, pp. 44–59, 2025.
- [49] H. Mukherjee, P. Sreerama, A. Dhar, S. M. Obaidullah, K. Roy, M. Mahmud, and K. Santosh, “Automatic lung health screening using respiratory sounds,” *Journal of Medical Systems*, vol. 45, no. 2, p. 19, 2021.
- [50] D. Bardou, K. Zhang, and S. M. Ahmad, “Lung sounds classification using convolutional neural networks,” *Artificial intelligence in medicine*, vol. 88, pp. 58–69, 2018.
- [51] M. Aykanat, Ö. Kılıç, B. Kurt, and S. Saryal, “Classification of lung sounds using convolutional neural networks,” *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, p. 65, 2017.
- [52] S. Yu, Y. Ding, K. Qian, B. Hu, W. Li, and B. W. Schuller, “A glance-and-gaze network for respiratory sound classification,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 9007–9011, IEEE, 2022.
- [53] L. Zhang and et al., “A feature polymerized based two-level ensemble model for respiratory sound classification,” in *Proc. BioCAS*, pp. 238–242, 2022.

- [54] E. Garmash and C. Monz, “Ensemble learning for multi-source neural machine translation,” in *Proc. COLING*, pp. 1409–1418, 2016.
- [55] M. M. Jaber, S. K. Abd, P. M. Shakeel, M. Burhanuddin, M. A. Mohammed, and S. Yussof, “A telemedicine tool framework for lung sounds classification using ensemble classifier algorithms,” *Measurement*, vol. 162, p. 107883, 2020.
- [56] B. Rocha and et al., “A respiratory sound database for the development of automated classification,” in *ICBHI Challenge*, pp. 33–37, 2018.
- [57] F.-S. Hsu, S.-R. Huang, C.-W. Huang, C.-J. Huang, Y.-R. Cheng, C.-C. Chen, J. Hsiao, C.-W. Chen, L.-C. Chen, Y.-C. Lai, *et al.*, “Benchmarking of eight recurrent neural network variants for breath phase and adventitious sound detection on a self-developed open-access lung sound database—hf_lung_v1,” *PLoS One*, vol. 16, no. 7, p. e0254134, 2021.
- [58] F.-S. Hsu, C.-J. Huang, C.-Y. Kuo, S.-R. Huang, Y.-R. Cheng, J.-H. Wang, Y.-L. Wu, T.-L. Tzeng, and F. Lai, “Development of a respiratory sound labeling software for training a deep learning-based respiratory sound analysis model,” in *International Forum on Medical Imaging in Asia 2021*, vol. 11792, pp. 109–114, SPIE, 2021.
- [59] C. Jácome, J. Ravn, E. Holsbø, J. C. Aviles-Solis, H. Melbye, and L. Ailo Bongo, “Convolutional neural network for breathing phase detection in lung sounds,” *Sensors*, vol. 19, no. 8, p. 1798, 2019.
- [60] Q. Zhang and et al., “Sprsound: Open-source sjtu paediatric respiratory sound database,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 16, no. 5, pp. 867–881, 2022.
- [61] N. K. Sharma, S. R. Chetupalli, D. Bhattacharya, D. Dutta, P. Mote, and S. Ganapathy, “The second dicova challenge: Dataset and performance

- analysis for diagnosis of covid-19 using acoustics,” in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 556–560, IEEE, 2022.
- [62] F. Owens and M. Murphy, “A short-time fourier transform,” *Signal Processing*, vol. 14, no. 1, pp. 3–10, 1988.
- [63] S. Sanei, D. Jarchi, and A. G. Constantinides, *Body sensor networking, design and algorithms*. John Wiley & Sons, 2020.
- [64] F. J. Harris, “On the use of windows for harmonic analysis with the discrete fourier transform,” *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 2005.
- [65] I. V. McLoughlin, *Speech and audio processing: a MATLAB-based approach*. Cambridge University Press, 2016.
- [66] McFee, Brian, R. Colin, L. Dawen, D. PW.Ellis, M. Matt, B. Eric, and N. Oriol, “librosa: Audio and music signal analysis in python,” in *Proc. of The 14th Python in Science Conference*, pp. 18–25, 2015.
- [67] J. C. Brown and M. S. Puckette, “An efficient algorithm for the calculation of a constant q transform,” *The Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698–2701, 1992.
- [68] R. D. Patterson, “Auditory filters and excitation patterns as representations of frequency resolution,” *Frequency selectivity in hearing*, 1986.
- [69] B. R. Glasberg and B. C. Moore, “Derivation of auditory filter shapes from notched-noise data,” *Hearing research*, vol. 47, no. 1-2, pp. 103–138, 1990.
- [70] D. P. Ellis, “Gammatone-like spectrograms.” webresource:<http://www.ee.columbia.edu/dpwe/resources/matlab/gammatonegram>, 2009.

- [71] J. Salamon and J. P. Bello, “Deep convolutional neural networks and data augmentation for environmental sound classification,” *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 279–283, 2017.
- [72] A. Rakotomamonjy and G. Gasso, “Histogram of gradients of time–frequency representations for audio scene classification,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 142–153, 2014.
- [73] A. Mesaros, T. Heittola, and T. Virtanen, “TUT database for acoustic scene classification and sound event detection,” in *24th European Signal Processing Conference 2016 (EUSIPCO 2016)*, (Budapest, Hungary), 2016.
- [74] A. Mesaros, T. Heittola, A. Diment, B. Elizalde, A. Shah, E. Vincent, B. Raj, and T. Virtanen, “Dcase 2017 challenge setup: Tasks, datasets and baseline system,” 2017.
- [75] A. Mesaros, T. Heittola, and T. Virtanen, “A multi-device dataset for urban acoustic scene classification,” in *Proc.DCASE*, pp. 9–13, November 2018.
- [76] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [77] S. Kullback and R. A. Leibler, “On information and sufficiency,” *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [78] A. Mesaros, T. Heittola, E. Benetos, P. Foster, M. Lagrange, T. Virtanen, and M. D. Plumbley, “Detection and classification of acoustic scenes and events: Outcome of the DCASE 2016 challenge,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, pp. 379–393, Feb 2018.

- [79] Z. Li, L. Zhang, S. Du, and W. Liu, “Acoustic scene classification based on binaural deep scattering spectra with cnn and lstm,” in *Proc. DCASE*, 2018.
- [80] Y. Li, X. Li, and Y. Zhang, “The seie-scut systems for challenge on dcase 2018: Deep learning techniques for audio representation and classification,” *Proc. DCASE*, 2018.
- [81] S. Zhao, T. N. T. Nguyen, W.-S. Gan, and D. L. Jones, “Adsc submission for dcase 2017: Acoustic scene classification using deep residual convolutional neural networks,” *DCASE2017 Challenge, Tech. Rep.*, 2017.
- [82] J. L. Dai Wei, P. Pham, S. Das, and S. Qu, “Acoustic scene recognition with deep neural networks (dcase challenge 2016),” *Robert Bosch Research and Technology Center*, vol. 3, 2016.
- [83] V. Bisot, S. Essid, and G. Richard, “Hog and subband power distribution image features for acoustic scene classification,” in *23rd European Signal Processing Conference (EUSIPCO)*, pp. 719–723, IEEE, 2015.
- [84] J. Tchorz and M. Weg, “Combination of amplitude modulation spectrogram features and mfccs for acoustic scene classification,” tech. rep., DCASE2018 Challenge, Tech. Rep, 2018.
- [85] J.-w. Jung, H.-s. Heo, H.-j. Shim, and H. Yu, “Dnn based multi-level features ensemble for acoustic scene classification,” *Tech. Rep., DCASE 2018 Challenge*, 2018.
- [86] J.-W. Jung, H.-S. Heo, I. Yang, S.-H. Yoon, H.-J. Shim, and H.-J. Yu, “Dnn-based audio scene classification for dcase 2017: dual input features, balancing cost, and stochastic data duplication,” *System*, vol. 4, no. 5, 2017.

- [87] S. H. Bae, I. Choi, and N. S. Kim, “Acoustic scene classification using parallel combination of lstm and cnn,” in *Proc. DCASE*, pp. 11–15, 2016.
- [88] J. Ye, T. Kobayashi, M. Murakawa, and T. Higuchi, “Acoustic scene classification based on sound textures and events,” in *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 1291–1294, 2015.
- [89] Q. Kong, T. Iqbal, Y. Xu, W. Wang, and M. D. Plumbley, “Dcase 2018 challenge survey cross-task convolutional neural network baseline,” *arXiv preprint arXiv:1808.00773*, 2018.
- [90] J. Wang, “Dcase 2018 task 1a: Acoustic scene classification by bi-lstm-cnn-net multichannel fusion,”
- [91] K. J. Piczak, “The details that matter: Frequency resolution of spectrograms in acoustic scene classification,” *Proc. DCASE*, pp. 103–107, 2017.
- [92] J. Kim and K. Lee, “Empirical study on ensemble method of deep neural networks for acoustic scene classification,” *Proc. DCASE*, 2016.
- [93] H. Phan, L. Hertel, M. Maass, P. Koch, and A. Mertins, “Label tree embeddings for acoustic scene classification,” in *Proceedings of the 24th ACM international conference on Multimedia*, pp. 486–490, ACM, 2016.
- [94] J. Wang and S. Li, “Self-attention mechanism based system for dcase2018 challenge task1 and task4,” in *Proc. DCASE Challenge*, pp. 1–5, 2018.
- [95] C. Roletscheck, T. Watzka, A. Seiderer, D. Schiller, and E. André, “Using an evolutionary approach to explore convolutional neural networks for acoustic scene classification,” *Proc. DCASE*, vol. 2018, pp. 158–162, 2018.
- [96] I. Kukanov, V. Hautamäki, and K. A. Lee, “Recurrent neural network and maximal figure of merit for acoustic event detection,” *Proc. DCASE*, 2017.

- [97] G. Takahashi, T. Yamada, S. Makino, and N. Ono, “Acoustic scene classification using deep neural network and frame-concatenated acoustic feature,” *Proc. DCASE*, 2016.
- [98] Y. Yin, R. R. Shah, and R. Zimmermann, “Learning and fusing multimodal deep features for acoustic scene categorization,” in *Proc. ACM*, pp. 1892–1900, ACM, 2018.
- [99] S. Waldekar and G. Saha, “Wavelet-based audio features for acoustic scene classification,” *Tech. Rep., DCASE2018 Challenge*, 2018.
- [100] L. Zhang and J. Han, “Acoustic scene classification using multi-layered temporal pooling based on deep convolutional neural network,” *Proc. DCASE*, 2018.
- [101] S. Park, S. Mun, Y. Lee, and H. Ko, “Acoustic scene classification based on convolutional neural network using double image features,” in *Proc. DCASE*, pp. 1–5, 2017.
- [102] B. Elizalde, A. Kumar, A. Shah, R. Badlani, E. Vincent, B. Raj, and I. Lane, “Experiments on the dcase challenge 2016: Acoustic scene classification and sound event detection in real life recording,” *arXiv preprint arXiv:1607.06706*, 2016.
- [103] H. Phan, L. Hertel, M. Maass, P. Koch, R. Mazur, and A. Mertins, “Improved audio scene classification based on label-tree embeddings and convolutional neural networks,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, pp. 1278–1290, 2017.
- [104] Z. Ren, Q. Kong, J. Han, M. D. Plumbley, and B. W. Schuller, “Attention-based atrous convolutional neural networks: Visualisation and understanding perspectives of acoustic scenes,” in *Proc. ICASSP*, pp. 56–60, IEEE, 2019.

- [105] B. Lehner, H. Eghbal-Zadeh, M. Dorfer, F. Korzeniowski, K. Koutini, and G. Widmer, “Classifying short acoustic scenes with i-vectors and cnns: Challenges and optimisations for the 2017 dcase asc task,” *Proc. DCASE*, 2017.
- [106] M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, “Dcase 2016 acoustic scene classification using convolutional neural networks,” in *Proc. Workshop Detection Classif. Acoust. Scenes Events*, pp. 95–99, 2016.
- [107] J. Ye, T. Kobayashi, N. Toyama, H. Tsuda, and M. Murakawa, “Acoustic scene classification using efficient summary statistics and multiple spectro-temporal descriptor fusion,” *Applied Sciences*, vol. 8, no. 8, p. 1363, 2018.
- [108] T. Nguyen and F. Pernkopf, “Acoustic scene classification using a convolutional neural network ensemble and nearest neighbor filters,” in *Proc. DCASE*, 2018.
- [109] A. Dang, T. Vu, and J.-C. Wang, “Acoustic scene classification using ensemble of convnets,” tech. rep., Tech. Rep., DCASE2018 Challenge, 2018.
- [110] R. Hyder, S. Ghaffarzadegan, Z. Feng, and T. Hasan, “Buet bosch consortium (b2c) acoustic scene classification systems for dcase 2017,” *Proc. DCASE*, 2017.
- [111] E. Marchi, D. Tonelli, X. Xu, F. Ringeval, J. Deng, and B. Schuller, “The up system for the 2016 dcase challenge using deep recurrent neural network and multiscale kernel subspace learning,” *Proc. DCASE*, 2016.
- [112] H. Phan, O. Y. Chén, P. Koch, L. Pham, I. McLoughlin, A. Mertins, and M. De Vos, “Beyond equal-length snippets: How long is sufficient to recognize an audio scene?,” *arXiv preprint arXiv:1811.01095*, 2018.

- [113] O. Mariotti, M. Cord, and O. Schwander, “Exploring deep vision models for acoustic scene classification,” *Proc. DCASE*, pp. 103–107, 2018.
- [114] Z. Weiping, Y. Jiantao, X. Xiaotao, L. Xiangtao, and P. Shaohu, “Acoustic scene classification using deep convolutional neural network and multiple spectrograms fusion,” *Proc. DCASE*, 2017.
- [115] S. Park, S. Mun, Y. Lee, and H. Ko, “Score fusion of classification systems for acoustic scene classification,” *Proc. DCASE*, 2016.
- [116] T. Zhang, K. Zhang, and J. Wu, “Multi-modal attention mechanisms in lstm and its application to acoustic scene classification.,” in *Interspeech*, pp. 3328–3332, 2018.
- [117] L. Yang, X. Chen, and L. Tao, “Acoustic scene classification using multi-scale features,” in *Workshop on DCASE*, 2018.
- [118] Y. Han, J. Park, and K. Lee, “Convolutional neural networks with binaural representations and background subtraction for acoustic scene classification,” *Proc. DCASE*, pp. 1–5, 2017.
- [119] V. Bisot, R. Serizel, S. Essid, and G. Richard, “Supervised nonnegative matrix factorization for acoustic scene classification,” *Proc. DCASE*, pp. 62–69, 2016.
- [120] T. Zhang, K. Zhang, and J. Wu, “Temporal transformer networks for acoustic scene classification.,” in *Interspeech*, pp. 1349–1353, 2018.
- [121] A. Golubkov and A. Lavrentyev, “Acoustic scene classification using convolutional neural networks and different channels representations and its fusion,” *Technical report, DCASE2018 Challenge*, 2018.

- [122] S. Mun, S. Park, D. K. Han, and H. Ko, “Generative adversarial network based acoustic scene training set augmentation and selection using svm hyper-plane,” *Proc. DCASE*, pp. 93–97, 2017.
- [123] H. Eghbal-Zadeh, B. Lehner, M. Dorfer, and G. Widmer, “Cp-jku submissions for dcase-2016: A hybrid approach using binaural i-vectors and deep convolutional neural networks,” *Proc. DCASE*, 2016.
- [124] H. Phan, O. Y. Chén, L. Pham, P. Koch, M. De Vos, I. McLoughlin, and A. Mertins, “Spatio-temporal attention pooling for audio scene classification,” *arXiv preprint arXiv:1904.03543*, 2019.
- [125] X. Bai, J. Du, Z.-R. Wang, and C.-H. Lee, “A hybrid approach to acoustic scene classification based on universal acoustic models,” *Proc. Interspeech 2019*, pp. 3619–3623, 2019.
- [126] Z. Ren, K. Qian, Y. Wang, Z. Zhang, V. Pandit, A. Baird, and B. Schuller, “Deep scalogram representations for acoustic scene classification,” *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 3, pp. 662–669, 2018.
- [127] S. Mun, S. Shon, W. Kim, D. K. Han, and H. Ko, “Deep neural network based learning and transferring mid-level audio features for acoustic scene classification,” in *Proc. ICASSP*, pp. 796–800, IEEE, 2017.
- [128] T. Nguyen and F. Pernkopf, “Acoustic scene classification with mismatched recording devices using mixture of experts layer,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1666–1671, IEEE, 2019.
- [129] L. Gao, H. Mi, B. Zhu, D. Feng, Y. Li, and Y. Peng, “An adversarial feature distillation method for audio classification,” *IEEE Access*, vol. 7, pp. 105319–105330, 2019.

- [130] Y. Yang, H. Zhang, W. Tu, H. Ai, L. Cai, R. Hu, and F. Xiang, “Kullback–leibler divergence frequency warping scale for acoustic scene classification using convolutional neural network,” in *Proc. ICASSP*, pp. 840–844, IEEE, 2019.
- [131] J. Li, W. Dai, F. Metze, S. Qu, and S. Das, “A comparison of deep learning methods for environmental sound detection,” in *Proc. ICASSP*, pp. 126–130, IEEE, 2017.
- [132] T. Nguyen and F. Pernkopf, “Acoustic scene classification with mismatched recording devices using mixture of experts layer,” in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1666–1671, IEEE, 2019.
- [133] R. Hyder, S. Ghaffarzadegan, Z. Feng, J. H. Hansen, and T. Hasan, “Acoustic scene classification using a cnn-supervector system trained with auditory and spectrogram image features.,” in *Interspeech*, pp. 3073–3077, 2017.
- [134] L. Yang, L. Tao, X. Chen, and X. Gu, “Multi-scale semantic feature fusion and data augmentation for acoustic scene classification,” *Applied Acoustics*, vol. 163, p. 107238, 2020.
- [135] S. S. R. Phaye, E. Benetos, and Y. Wang, “Subspectralnet—using sub-spectrogram based convolutional neural networks for acoustic scene classification,” in *Proc. ICASSP*, pp. 825–829, IEEE, 2019.
- [136] Y. Wu and T. Lee, “Enhancing sound texture in cnn-based acoustic scene classification,” in *Proc. ICASSP*, pp. 815–819, IEEE, 2019.
- [137] H. Song, J. Han, and S. Deng, “A compact and discriminative feature based on auditory summary statistics for acoustic scene classification,” *arXiv preprint arXiv:1904.05243*, 2019.

- [138] H.-S. Heo, J.-w. Jung, H.-j. Shim, and H.-J. Yu, “Acoustic scene classification using teacher-student learning with soft-labels,” *arXiv preprint arXiv:1904.10135*, 2019.
- [139] H. Chen, P. Zhang, and Y. Yan, “An audio scene classification framework with embedded filters and a dct-based temporal module,” in *Proc. ICASSP*, pp. 835–839, 2019.
- [140] M. Akay, “Wavelet applications in medicine,” *IEEE spectrum*, vol. 34, no. 5, pp. 50–56, 1997.
- [141] L. Jonathan and O. Sofia, “Higher-order properties of analytic wavelets,” *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 146–160, 2008.
- [142] J.-C. Hong and Y. Kim, “Determination of the optimal gabor wavelet shape for the best time-frequency localization using the entropy concept,” *Experimental mechanics*, vol. 44, no. 4, pp. 387–395, 2004.
- [143] L. Li, H. Cai, and Q. Jiang, “Adaptive synchrosqueezing transform with a time-varying parameter for non-stationary signal separation,” *Applied and Computational Harmonic Analysis*, vol. 49, no. 3, pp. 1075–1106, 2020.
- [144] M. Misiti, Y. Misiti, G. Oppenheim, and J.-M. Poggi, “Wavelet toolbox,” *The MathWorks Inc., Natick, MA*, vol. 15, p. 21, 1996.
- [145] M. Holschneider, *Wavelets: an analysis tool*. Oxford University Press, 1995.
- [146] B. Rocha, D. Filos, L. Mendes, Vogiatzis, *et al.*, “A respiratory sound database for the development of automated classification,” in *Precision Medicine Powered by pHealth and Connected Health*, pp. 33–37, 2018.
- [147] D. Perna, “Convolutional neural networks learning from respiratory data,” in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 2109–2113, IEEE, 2018.

- [148] B. M. Rocha, D. Filos, L. Mendes, G. Serbes, S. Ulukaya, Y. P. Kahya, N. Jakovljevic, T. L. Turukalo, I. M. Vogiatzis, E. Perantoni, *et al.*, “An open access database for the evaluation of respiratory sound classification algorithms,” *Physiological measurement*, vol. 40, no. 3, p. 035001, 2019.
- [149] N. Jakovljević and T. Lončar-Turukalo, “Hidden markov model based respiratory sound classification,” in *Precision Medicine Powered by pHealth and Connected Health*, pp. 39–43, Springer, 2018.
- [150] G. Serbes, S. Ulukaya, and Y. P. Kahya, “An automated lung sound pre-processing and classification system based on spectral analysis methods,” in *Precision Medicine Powered by pHealth and Connected Health*, pp. 45–49, Springer, 2018.
- [151] L. Pham, H. Phan, R. King, A. Mertins, and I. McLoughlin, “Inception-based network and multi-spectrogram ensemble applied for predicting respiratory anomalies and lung diseases,” in *Proc. EMBC*, pp. 253–256, 2021.
- [152] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “Panns: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.
- [153] D. S. Park, W. Chan, Y. Zhang, C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” *preprint arXiv:1904.08779*, 2019.
- [154] K. Xu, D. Feng, H. Mi, B. Zhu, D. Wang, L. Zhang, H. Cai, and S. Liu, “Mixup-based acoustic scene classification using multi-channel convolutional neural network,” in *Pacific Rim Conference on Multimedia*, pp. 14–23, 2018.
- [155] Y. Tokozume, Y. Ushiku, and T. Harada, “Learning from between-class examples for deep sound recognition,” in *ICLR*, 2018.

- [156] S. I. and C. S., “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proc. ICML*, pp. 448–456, 2015.
- [157] V. Nair and E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proc. ICML*, 2010.
- [158] N. Srivastava and et al., “Dropout: a simple way to prevent neural networks from overfitting,” *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [159] F. Chollet, “Keras library.” <https://keras.io>, 2015.
- [160] P. K. Diederik and B. Jimmy, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2015.
- [161] I. . Challenge. https://bhichallenge.med.auth.gr/sites/default/files/ICBHI_final_database/ICBHI_challenge_train_test.txt.
- [162] B. Rocha *et al.*, “A respiratory sound database for the development of automated classification,” in *Precision Medicine Powered by pHealth and Connected Health*, pp. 33–37, 2018.
- [163] P. Faustino, J. Oliveira, and M. Coimbra, “Crackle and wheeze detection in lung sound signals using convolutional neural networks,” in *Proc. EMBC*, pp. 345–348, 2021.
- [164] Y. Ma, X. Xu, Q. Yu, Y. Zhang, Y. Li, J. Zhao, and G. Wang, “Lungbrn: A smart digital stethoscope for detecting respiratory disease using bi-resnet deep learning algorithm,” in *Proc. BioCAS*, pp. 1–4, 2019.
- [165] Y. Ma, X. Xu, and Y. Li, “LungRN+NL: An improved adventitious lung sound classification using non-local block resnet neural network with mixup data augmentation,” in *Proc. INTERSPEECH*, pp. 2902–2906, 2020.

- [166] S. Gairola, F. Tom, N. Kwatra, and M. Jain, “Respirenet: A deep neural network for accurately detecting abnormal lung sounds in limited data setting,” in *Proc. EMBC*, pp. 527–530, 2021.
- [167] W. Ellis, “Gammatone-like spectrogram,” 2009.
- [168] Y. Tokozume and et al., “Learning from between-class examples for deep sound recognition,” *arXiv preprint arXiv:1711.10282*, 2017.
- [169] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “SpecAugment: A simple data augmentation method for automatic speech recognition,” *arXiv preprint arXiv:1904.08779*, 2019.
- [170] B. Kim and et al., “Qti submission to dcase 2021: Residual normalization for device-imbalanced acoustic scene classification with efficient design,” *arXiv preprint arXiv:2206.13909*, 2022.
- [171] C. Szegedy and et al., “Going deeper with convolutions,” in *Proc. CVPR*, pp. 1–9, 2015.
- [172] L. Pham and et al., “Inception-based network and multi-spectrogram ensemble applied to predict respiratory anomalies and lung diseases,” in *Proc. EMBC*, pp. 253–256, 2021.
- [173] A. Vaswani and et al., “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [174] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, pp. 539–546, IEEE, 2005.
- [175] J. Li and et al., “Improving the resnet-based respiratory sound classification systems with focal loss,” in *Proc. BioCAS*, pp. 223–227, 2022.

- [176] Z. Chen and et al., “Classify respiratory abnormality in lung sounds using stft and a fine-tuned resnet18 network,” in *Proc. BioCAS*, pp. 233–237, 2022.
- [177] L. Pham and et al., “Wider or deeper neural network architecture for acoustic scene classification with mismatched recording devices,” in *Proc. ACM*, pp. 1–5, 2022.
- [178] P. Muguli and et al., “Dicova challenge: Dataset, task, and baseline system for covid-19 diagnosis using acoustics,” *preprint arXiv:2103.09148*, 2021.
- [179] T. S. D. Challenge, “Leaderboard results.” <https://competitions.codalab.org/competitions/34801#results>, 2021-10-02.
- [180] J. F. Gemmeke and et al., “Audio set: An ontology and human-labeled dataset for audio events,” in *Proc. ICASSP*, pp. 776–780, 2017.
- [181] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, “Object recognition with gradient-based learning,” in *Shape, contour and grouping in computer vision*, pp. 319–345, Springer, 1999.
- [182] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *Proc. CVPR*, pp. 1251–1258, 2017.
- [183] C. Szegedy and et al., “Rethinking the inception architecture for computer vision,” in *Proc. CVPR*, pp. 2818–2826, 2016.
- [184] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proc. PMLR*, pp. 448–456, 2015.
- [185] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *preprint arXiv:1409.1556*, 2014.

- [186] A. G. Howard and et al., “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *preprint arXiv:1704.04861*, 2017.
- [187] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. CVPR*, pp. 770–778, 2016.
- [188] G. Huang and et al., “Densely connected convolutional networks,” in *Proc. CVPR*, pp. 4700–4708, 2017.
- [189] J.Cramer and et al., “Look, listen, and learn more: Design choices for deep audio embeddings,” in *Proc. ICASSP*, pp. 3852–3856, 2019.
- [190] J. Shor and et al., “Towards learning a universal non-semantic representation of speech,” *preprint arXiv:2002.12764*, 2020.
- [191] K.Guolin and et al., “Lightgbm: A highly efficient gradient boosting decision tree,” *Advances in neural information processing systems*, vol. 30, pp. 3146–3154, 2017.
- [192] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, *et al.*, “{TensorFlow}: a system for {Large-Scale} machine learning,” in *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pp. 265–283, 2016.
- [193] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshein, L. Antiga, *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” *Advances in neural information processing systems*, vol. 32, 2019.
- [194] N.Hien, E.Cooper, and K. Kamei, “Borderline over-sampling for imbalanced data classification,” *International Journal of Knowledge Engineering and Soft Data Paradigms*, vol. 3, no. 1, pp. 4–21, 2011.

- [195] F. Pedregosa and et al., “Scikit-learn: Machine learning in python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
- [196] Q. Gao, J. Liu, H. Zhang, J. Hou, and X. Yang, “Enhanced fisher discriminant criterion for image recognition,” *Pattern Recognition*, vol. 45, no. 10, pp. 3717–3724, 2012.
- [197] Q. Zhang, J. Zhang, J. Yuan, H. Huang, Y. Zhang, C. Chen, J. Lin, B. Zhang, G. Lv, S. Lin, *et al.*, “Grand challenge on respiratory sound classification for sprsound dataset,” in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–5, IEEE, 2023.
- [198] J. Hu, C. S. Leow, S. Tao, W. L. Goh, and Y. Gao, “Supervised contrastive pretrained resnet with mixup to enhance respiratory sound classification on imbalanced and limited dataset,” in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–5, IEEE, 2023.
- [199] D. Pessoa, G. Petmezas, V. E. Papageorgiou, B. M. Rocha, L. Stefanopoulos, V. Kilintzis, N. Maglaveras, I. Frerichs, P. de Carvalho, and R. P. Paiva, “Pediatric respiratory sound classification using a dual input deep learning architecture,” in *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, pp. 1–5, IEEE, 2023.