Supervised Contrastive Learning Framework and Hardware Implementation of Learned ResNet for Real-time Respiratory Sound Classification

Jinhai Hu, *Graduate Student Member, IEEE*, Cong Sheng Leow, *Graduate Student Member, IEEE*, Shuailin Tao, *Graduate Student Member, IEEE*, Wang Ling Goh, *Senior Member, IEEE*, and Yuan Gao, *Member, IEEE*

***Abstract*—This paper presents a supervised contrastive learning (SCL) framework for respiratory sound classification and the hardware implementation of learned ResNet on field programmable gate array (FPGA) for real-time monitoring. At the algorithmic level, multiple techniques such as feature augmentation and MixUp are combined holistically to mitigate the impact of data scarcity and imbalanced classes in the training dataset. Bayesian optimization further enhances the classification accuracy through parameter tuning in pre-processing and SCL. The proposed framework achieves 0.8725 total score (including runtime score) on a ResNet-18 model in both event and record multi-class classification tasks using the SJTU Paediatric Respiratory Sound Database (SPRSound). In addition, algorithm-hardware co-optimizations including Quantization-Aware Training (QAT), merge of network layers, optimization of memory size and number of parallel threads are performed for hardware implementation on FPGA. This approach reduces 40% model size and 70% computation latency. The learned ResNet is implemented on a Xilinx Zynq ZCU102 FPGA with 16ms latency and less than 2% inference score degradation compared to the software model.**

***Index Terms*—Respiratory sound classification, Balanced sampler, Supervised contrastive learning, MixUp finetuning, Bayesian optimization, Fixed-point quantization, FPGA.**

I. Introduction

Respiratory diseases such as chronic obstructive pulmonary disease (COPD) and lower respiratory infections are the leading causes of death globally [1]. Early detection of respiratory disorders helps clinician to diagnose and intervene timely for better treatment outcome [2]. Auscultation with stethoscope is the primary diagnosis tool used by clinician to identify abnormal respiratory sound [3]. The current medical practice requires a clinician to conduct the auscultation procedure. Due to the varying individual auditory systems, the human factor may affect the diagnosis results. To address the accessibility, reliability, and consistency challenges in the diagnosis of respiratory diseases, automated respiratory sound classification has attracted considerable research interests. Early efforts used machine learning methods such as support vector machine (SVM) [4], Hidden Markov Model [5], Naïve Bayes [6], and Logistic Regression [7]. More recent works focus on deep neural networks [8-17]. These methods convert the one-dimensional time-domain respiratory sound data to two-dimensional array using spectral transformations such as log-Mel spectrogram, short-term Fourier transform (STFT), Mel coefficients (MFCC), and wavelet transformation. Deep neural networks such as CNN, RNN, attention-based algorithm and hybrid neural networks are then used to extract and classify the feature representations [18-20].

The accuracy and robustness of AI algorithm are highly dependent on the training data quality and the accuracy of labeling. Common limitations of the existing respiratory sound databases are data scarcity and class imbalance [21]. Most labels show normal status in the publicly available databases such as ICBHI [22] and SJTU paediatric respiratory sound database (SPRSound) [21]. Imbalanced datasets lead to biases where the trained neural network favours the majority class. As most of the minority classes indicate abnormal events, the biases against the minority classes result in significant false negatives. The robustness of a learning algorithm can be enhanced through data augmentation, model pruning and parameter penalty. The efficacy of the method can be evaluated using various datasets [23], [24]. Online learning also can be applied to finetune model parameters with actual data to improve robustness [25], [26].

In recent years, contrastive learning has emerged as a powerful technique for representation learning [27]. It has

Manuscript received 16 Jan. 2024; revised 17 Mar. 2024 and 04 May 2024; accepted 26 May 2024. This work was supported by the Agency for Science, Technology and Research (A*STAR), Singapore under the Nanosystems at the Edge programme, grant No. A18A1b0055. Jinhai Hu and Cong Sheng Leow contributed equally to this work. (Corresponding author: Yuan Gao)

J. Hu is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 and he is also with Institute of Microelectronics (IME), A*STAR, Singapore 138634 (e-mail: jinhai001@e.ntu.edu.sg).

C. S. Leow was with the Institute of Microelectronics (IME), A*STAR, Singapore 138634. He is now with the Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, MI 48109 USA (email: leowcs@umich.edu).

S. Tao is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 and he is also with Institute of Microelectronics (IME), A*STAR, Singapore 138634 (e-mail: shuailin001@e.ntu.edu.sg).

W. L. Goh is with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: ewlgoh@ntu.edu.sg).

Y. Gao is with the Institute of Microelectronics (IME), A*STAR, Singapore 138634 (e-mail: gaoy@ime.a-star.edu.sg).

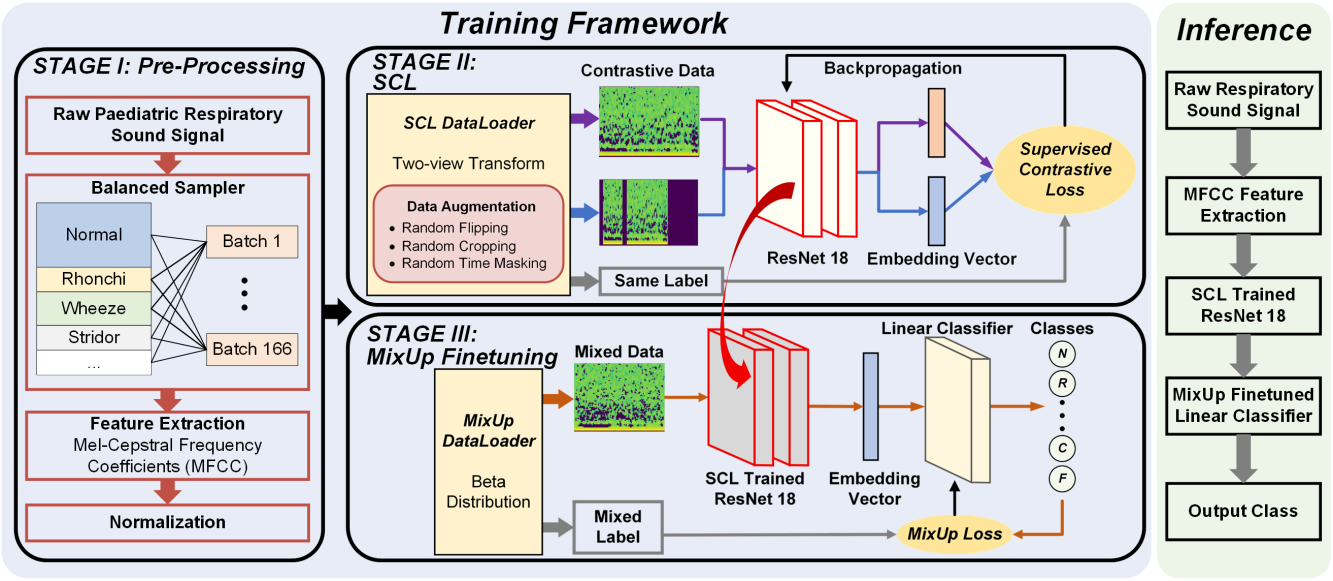

Fig. 1. (left) Block diagram of training framework, including 3 stages of pre-processing, SCL and MixUp finetuning. (right) Block diagram of model for inference.

shown remarkable success in unsupervised training of deep image models [28] and sound events learning [29], [30]. Recently, the introduction of supervised contrastive learning (SCL) allows better utilization of labelled data, and it is proven effective in preventing overfitting [31]. Hence, SCL has great potential in respiratory sound classification to overcome the challenge of data scarcity and class imbalance. On the other hand, to enable continuous monitoring outside hospital setting, the algorithm should be suitable for implementation in portable devices. Most of the recently reported bio-signal monitoring hardware designs aims for cardiac monitoring [32], electroencephalogram (EEG) monitoring [33] and cough detection [34]. There is very limited literature on hardware implementation for real-time respiratory sound classification.

In this work, we present a SCL framework for neural network training with limited and imbalanced dataset as well as the hardware implementation of the learned ResNet on field programmable gate array (FPGA) for real-time respiratory sound classification. The contributions of this work can be summarized as follows,

(1) The development of a SCL framework that combined multiple techniques in a holistic manner to address data imbalance and overfitting in the neural network training process. More specifically, a balanced sampler is used to generate more representations from the minority classes to reduce the bias. Two-view transform with random data augmentation enhances SCL model's generalization and robustness. MixUp is used to fine tune the model and prevent overfitting. In addition, Bayesian optimization further enhances the learning and classification accuracy through the tuning of hyperparameters and pre-processing parameters. These techniques effectively improve the accuracy and robustness of the respiratory sound classification algorithm.

(2) Algorithm-hardware co-optimization for FPGA implementation with low hardware cost and latency. By embedding weight quantization into the network training loop, the quantization loss can be effectively compensated during quantization-aware training. The quantized model is further optimized by merging batch normalization into convolution layer to reduce the number of MAC operations. The optimization of memory size and number of parallel threads reduces latency and improves data throughput. This co-optimization reduces model size by 40% and computation latency by 70%.

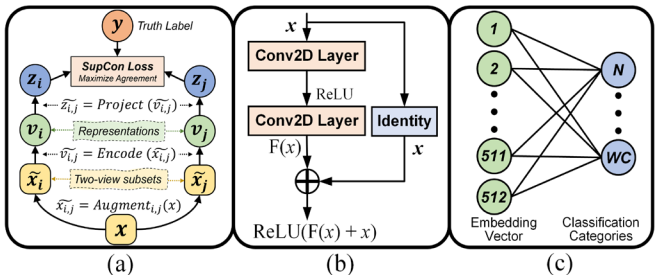

Fig. 2. (a) flow chart of SCL, (b) basic residual block in ResNet-18 network, (c) Fully connected layer linear classifier in SCL.

The proposed SCL framework achieved the best overall performance in the 2023 IEEE Biomedical Circuits and Systems Conference (BioCAS) Grand Challenge on respiratory sound classification using the SPRSound database [35]. This paper is an extended version of [36] with additional comprehensive algorithm analysis, details of hardware implementation and measurement results. The rest of this paper is organized as follows: Section II introduces the proposed SCL framework. The details of building blocks are presented in Section III. Inference and hardware implementation are elucidated in Section IV. Section V delves into the classification results, hardware utilization, and performance benchmarking. Finally, Section VI concludes the work.

II. Proposed Framework

Fig. 1(left) shows the block diagram of the proposed framework for model training. It contains three main stages:

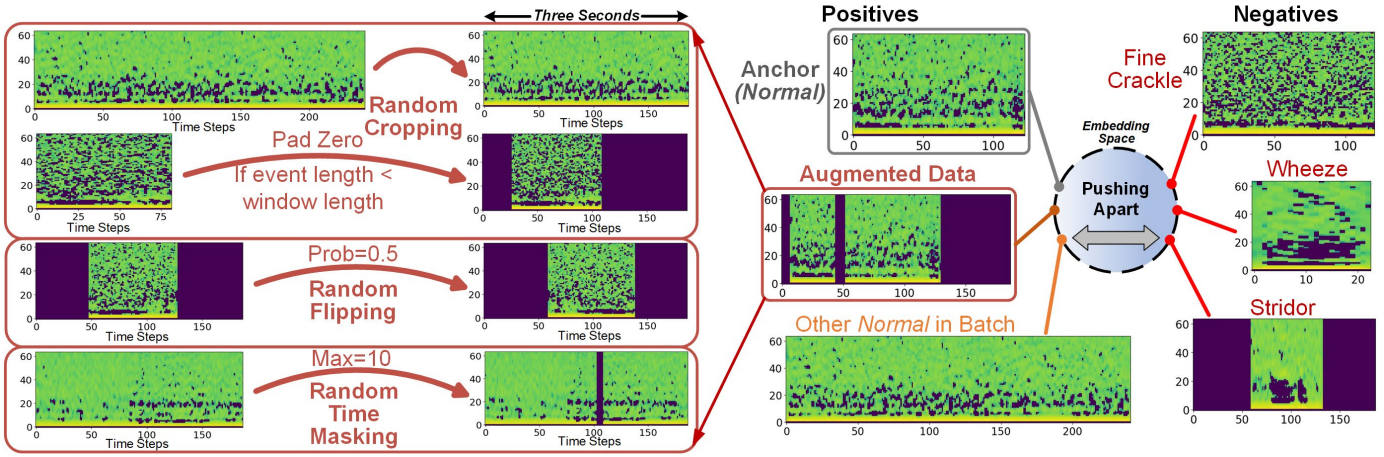

Fig. 3. (left) MFCC feature augmentation techniques for event classification task training. (right) Illustration of how SCL pushes apart positive and negative classes in the embedding space.

pre-processing, supervised contrastive learning and Mixup finetuning. The raw respiratory signal is firstly pre-processed by a balanced sampler to reduce the bias due to imbalanced dataset. Next, the input signal MFCC feature map is generated and standardized with Z-score normalization [37]. In Stage II, MFCC feature map is used to train the neural network using SCL. Data augmentation techniques are applied to MFCC feature map to generate two correlated views of the data. The SCL algorithm flow chart is shown in Fig. 2(a) (details refer to Section III C). In this work, ResNet-18 [38] is selected as the backbone of SCL for its relatively simple structure and low hardware resource requirement. The basic residual block in ResNet-18 is shown in Fig. 2(b). Guided by the supervised contrastive loss function (SupCon), embedded data with the same label are clustered together while data with different labels are clustered further away, enhancing model's generalization capability and robustness. The trained ResNet-18 network is transferred to Stage III for fine tuning. To mitigate the risk of overfitting, the MixUp method is employed during fine-tuning [39]. The embedded vectors are then used for classification using a fully connected layer linear classifier shown in Fig. 2(c).

The SPRSound database [21] is used in this work for model training and performance evaluation. It contains 6656 labelled events and 1949 labelled recordings. Each recording is segmented into multiple respiratory events, annotated as Normal (N, 77.5%), Wheeze (W, 6.8%), Rhonchi (R, 0.6%), Stridor (S, 0.2%), Coarse Crackle (CC, 0.7%), Fine Crackle (FC, 13.7%) or Wheeze & Crackle (W&C, 0.5%). Recordings are labelled as Normal (N, 66.9%), Continuous Adventitious Sounds (CAS, 6.9%), Discontinuous Adventitious Sounds (DAS, 12.7%), CAS & DAS (C&D, 4.4%) or Poor Quality (PQ, 9.1%). The audio files are mono-channel recorded and sampled at 8kHz with 16-bit data resoultion. The four classification tasks are:

Task 1-1: Binary classification of *Normal* and *Adventitious*;
Task 1-2: 7-class classification of *N, W, R, S, CC, FC, W&C*;
Task 2-1: 3-class classification of *Normal*, *Adventitious* and *Poor-Quality*;
Task 2-2: Five-class classification of N, *CAS, DAS, C&D, PQ*.

III. Stages of Proposed Framework

A. Data Pre-processing

Input voice data is firstly processed by time-domain normalization to scale the signal's amplitude to ±0.5 with zero mean. Next, MFCC feature map is generated with optimized hop length and number of filter channels (details refer to Section III.E). To ensure uniform input level to the following SCL model, Z-score normalization is performed to standardize the MFCC feature map.

With an intrinsically imbalanced dataset, a balanced sampler is required to reduce the classification bias towards the majority classes. By assigning each class a weight that is reverse proportional to the class's percentage in the dataset, this balanced sampler ensures that similar number of samples from each class can be selected in each data batch.

B. Data Augmentation

To generate more data representations from the limited dataset available for model training, data augmentation is required to generate subsets of the original training data. Inspired by the data augmentation techniques proposed in [40], random cropping, random flipping and random time masking are applied to the MFCC feature map as shown in Fig. 3 (left).

Random cropping serves two functions. Firstly, a uniform feature size can be obtained by either trimming away the data that exceeds the window size or padding the data shorter than the window size with the mean value obtained from Z-normalization. Secondly, random cropping emulates event shift along the temporal axis, enabling the network to develop better tolerance to temporal variations and shifts. Random flipping enables the network to learn and generalize from different directional patterns when data is presented in different orientations. Random time masking purposely obfuscates a randomly selected segment of data in the time window. This method ensures that a model doesn't over-rely on specific time segments and learns more generalized features.

To preserve the pathological features of the respiratory sounds, data augmentation is not carried out at the inference stage. At the training stage, it may appear that the pathological

feature of the data can potentially be corrupted based on the augmentations. However, it should be noted that the augmentation does not alter the signal but is merely providing an alternative "view" of the signal of the same label. While it is possible that augmentations such as random cropping can crop out the critical sections which led to the initial label being annotated, the augmented data is still derived from the original signal. This imply that there may be residual features from the critical section. While it is not the focus of this work, a rigorous mathematical study can be helpful in determining the presence of such residue and its impact. The random nature of the augmentations allows such augmentations to be analogous to noise to help improve robustness of the model.

C. Supervised Contrastive Learning

Fig. 2(a) shows the SCL algorithm flow chart. The input respiratory sound feature (x) undergoes data augmentation ($Augment_{i,j}(x)$) to generate two correlated views (x_i, x_j), which contain subset of information from the original sample. The contrastive representations (v_i, v_j) are generated through ResNet-18 as an encoder. The SupCon loss is calculated using embedding vectors (z_i, z_j) from the two-view transform together with the ground truth label. Thus, the process to reduce SupCon loss in model training also minimize the distance among different representations with the same type of label, so that they can be pulled together in the embedding space.

The contrastive loss function, denoted as L^{sup} as shown below in (1), governs the learning process.

$$L^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp\left(\frac{z_i \cdot z_p}{\tau}\right)}{\sum_{a \in A(i)} \exp\left(\frac{z_i \cdot z_a}{\tau}\right)} \qquad (1)$$

where i is the index of an arbitrary augmented sample within a training batch I, $P(i)$ is the set of indices of all positives in that batch except i, and $|P(i)|$ is its cardinality, z is the normalized embedded vector, τ is a scalar temperature parameter, and $\mathrm{A(i)} \equiv I \backslash \{i\}$. This loss is used to update the weights of the SCL model. Supervised by the label and the contrastive loss generated from two-view transform, clusters of points belonging to the same class are pulled together in the embedding space, while simultaneously pushing apart clusters of samples from different classes as illustrated in Fig. 3 (right).

D. MixUp Finetuning

The SCL model plays a crucial role in reducing the initial loss and improving the accuracy during the subsequent fine-tuning process. However, if the same dataset is used for both SCL training and finetuning, there is a high risk of overfitting, particularly when working with limited training datasets. To address this issue, the MixUp method is used to regularize the network with new data samples by blending original samples with the samples from other classes. The labels for the mixed samples are created based on a combination of their corresponding labels [39]. The MixUp process is represented as follows:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j \qquad (2)$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \qquad (3)$$

TABLE I
COMPARISON ON PERFORMANCE OF MIXUP METHOD

Topology	Dataset	Task 11	Task 12	Task 21	Task 22
Linear Classifier	Train	0.998	0.976	0.938	0.895
	Valid	0.898	0.854	0.785	0.708
	Test	0.828	0.821	0.746	0.607
+ MixUp	Train	0.950	0.924	0.874	0.836
	Valid	0.913	0.866	0.792	0.711
	Test	**0.849**	**0.840**	**0.763**	**0.629**

where λ is controlled by the MixUp interpolation coefficient α and follows a beta distribution $X \sim Beta(\alpha, \alpha)$. By training the model on these synthesized samples, the MixUp method allows the model to learn from the relationships and patterns in different samples. Although the MixUp method may lead to a slight degradation of training accuracy, our implementation shows improvements in both validation and test scores during the finetuning process, as presented in Table I.

E. Bayesian Optimization

To identify the optimal SCL model configuration, Bayesian optimization is employed using the open-source code *Tune* [41]. This optimization process involves the tuning of both the pre-processing parameters, including hop length, number of MFCC channel, FFT length and the hyperparameters, including learning rate, number of convolution layer filters, number of neurons. For SCL training, the main parameters included in the search space are the temperature and learning rate.

IV. HARDWARE IMPLEMENTATION

For hardware implementation on resource-constraint mobile devices, the software algorithms which are commonly trained in 32-bit floating point (FP32) precision need to be quantized to lower fixed bit-width. Furthermore, for practical application in diagnosis and daily monitoring, real-time classification is an essential feature.

A. Fixed-point Quantization

Both post-training quantization (PTQ) and quantization-aware training (QAT) [42] are evaluated to compare their performance. It should be noted that, for both methods, the input MFCC feature maps are fixed at INT8 for various quantization bit-widths to prevent excessive loss of information, while model weights, bias, and activations (interlayer outputs) are quantized to the respective fixed bit-width.

The PTQ function is implemented in Python as part of the proposed algorithm framework [43]. The trained FP32 weights are quantized to fixed-point integer numbers of signed 2×2^n states (s), where n denotes the bit length. The quantization procedure starts with the searching for the maximum and minimum numbers in the weight matrix, which are denoted as w_{max} and w_{min}, respectively. These values set the boundaries of states, and the step size is defined as

$$\Delta w = \frac{w_{max} - w_{min}}{2^n - 1} \qquad (4)$$

For each weight entry in the weight matrix, the mapping process follows (5),

$$w_j = s_{j+1}, \quad if\ |w_i - (s_1 + j \times \Delta w)| \leq \frac{\Delta w}{2} \qquad (5)$$

where $j = 0, 1, 2, \ldots, 2^n - 1$.

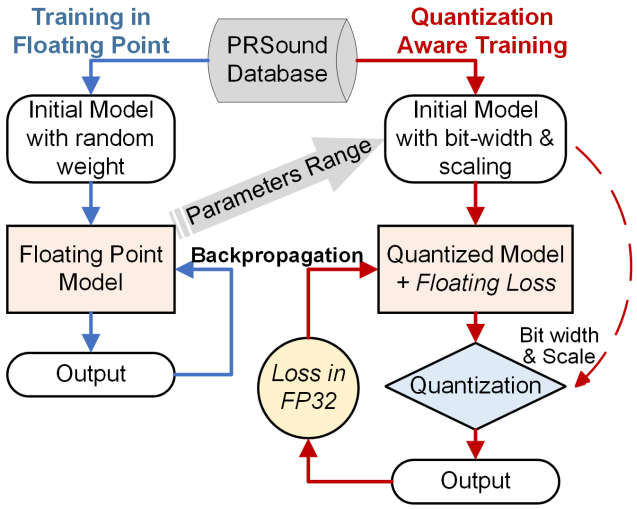

Fig. 4. The process flow of quantization aware training.

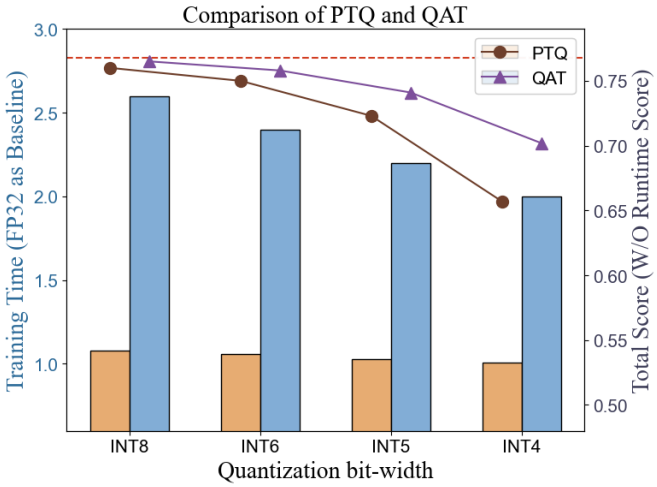

Fig. 5. Comparison of PTQ and QAT results, bar chart shows the training duration and line plot shows the total score.

The QAT flow chart is shown in Fig. 4. Unlike PTQ, the QAT process is performed within the training loop [44]. As shown in Fig. 4, the floating-point training is firstly initiated with random initial weights. Then, backpropagation computes the loss in FP32 format to guide the adjustment of model weights. The trained FP32 model sets the range of weights for QAT. To enable a direct memory mapping to FPGA, all the model parameters, encompassing the MFCC feature input, layer weights, biases, and activations (interlayer output), are restricted to radix-2 number format throughout the training process. In addition, Bayesian optimization helps for the tuning of hyperparameter such as learning rate to avoid falling into extreme values, which may cause missing intermediate quantization states or hindering weight adjustment between states.

The QAT process yields a quantized model accompanied with a quantization loss in FP32 format. Although the training time of QAT is longer than PTQ, the classification performance of QAT is better as the results is less sensitive to the reduction of bit-width from 32-bit (FP32) to 4-bit (INT4) as seen in Fig. 5.

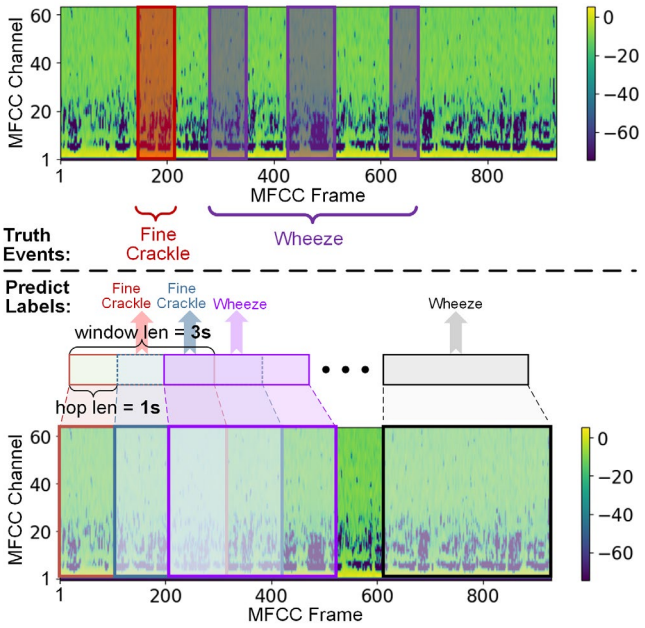

Fig. 6. Illustration of sliding-window based real-time respiratory sound signal recognition.

B. Real-time Inference

An illustration of the sliding-window based real-time respiratory sound inference process is shown in Fig. 6. The statistic characteristics of respiratory sound show that events usually have a duration between one second and three seconds [21]. Hence, the sliding window length is set as three seconds and each time step move one second, so that each feature will be captured by at least one window.

For real-time inference, the quantized model is retrained with data segmented in three-second length to match with the sliding window size. Another modification to the real-time inference algorithm is the number of classes for each task. Grand Challenge'23 only requires 2-class classification of "Normal" and "Adventitious" for Task 1-1, 7-class classification of "Normal" and 6 "Adventitious" events for Task 1-2, omitting the poor quality class in the SPRSound and Grand Challenge'23 datasets. However, this class is required in real-time inference for the scenario where the input signal quality is poor. Hence, the classification requires 3-class for real-time Task 1-1 and 8-class for real-time Task 1-2. The number of periods to be evaluated is calculated based on (6).

$$n = \left\lceil \frac{record\ length - window\ size}{sliding\ length} \right\rceil \tag{6}$$

The conversion between the period and number of frames in the feature map can be found in (7), where the sampling rate is 8 kHz, and the hop length is 128.

$$Frame = \frac{time(s) \times SamplingRate(Hz)}{hop\ length} \tag{7}$$

By applying the sliding-window technique, real-time inference can be achieved by continuously analysing the signal within the sliding window. This allows for the monitoring of the lung sound status in real-time to provide event outputs with minimal latency. Additionally, the occurrence of event-level

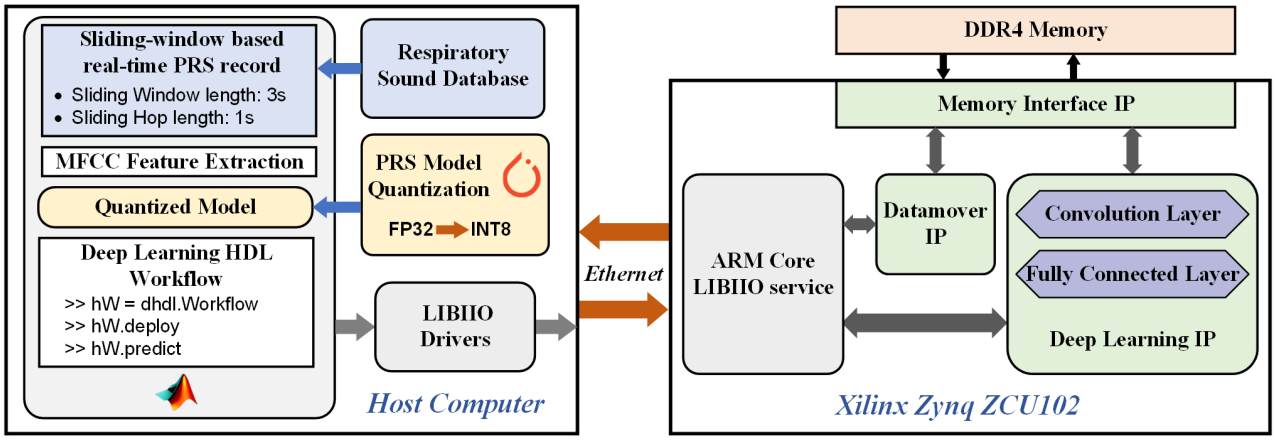

Fig. 7. High-level architecture for model implementation on Zynq ZCU102 FPGA using MATLAB Deep Learning HDL toolbox and existing IPs.

labels can be used to further cluster and analyse the record-level labels, providing valuable insights into the distribution and occurrence patterns of different events.

C. FPGA Implementation

Fig. 7 shows the high-level system architecture for model implementation on FPGA with the assistance of existing IPs. The quantized neural network model in Python is firstly converted to MATLAB code and then compiled to executable HDL using MATLAB Deep Learning HDL Toolbox [45]. As shown in Fig. 7, the ARM core communicates with the host using a Linux Industrial Input/ Output (LIBIIO) IP via Ethernet and accesses the DDR4 memory using Datamover IP. The ResNet-18 network is implemented with Deep Learning IP [46]. The key parameters for MFCC block include 64 MFCC channels, 512-point FFT and hop length of 128. The extracted feature maps are streamed to FPGA continuously, and the output of linear classifier is sent back to host to generate classification result.

MATLAB code optimizations are performed to further improve the FPGA inference performance as listed below:
(1) As FPGA only performs the inference task, the batch normalization layer can be merged into the convolution layer by embedding the mean and variance parameters used in batch normalization into the weight and bias parameters of convolution layer to reduce the number of multiply-accumulate (MAC) operations, leading to faster inference speed.
(2) Other than using the default size BRAM allocated by Deep Learning IP, the cache BRAM size for convolution operation is modified to match with the dimension of input MFCC feature map. Similarly, the cache BRAM size for fully connected layer also matches with the number of classes. Additionally, the number of parallel MAC threads for convolution operation and fully connected processor are optimized to 64 and 16 to streamline the process flow to reduce latency and in turn improve data throughput.

V. Results and Discussion

A. SCL Training Results

Fig. 8 shows the confusion matrix of four classification tasks using SPRSound testset. It indicates that the learned model performs better in event-level classification (Tasks 1-1 and 1-2) compared to record-level classification (Tasks 2-1 and 2-2), as records typically have longer durations and include more complex features. The model training results is visualized using t-SNE plots as shown in Fig. 9. The plots reveal distinct clustering according to the various sound types across different diagnostic tasks, demonstrating the SCL's exceptional ability to differentiate between the normal respiratory sounds and pathological respiratory sounds. These visualizations highlight the efficacy of the SCL embedding model in segregating complex audio features for potential use in both event and recording level automated respiratory diagnosis.

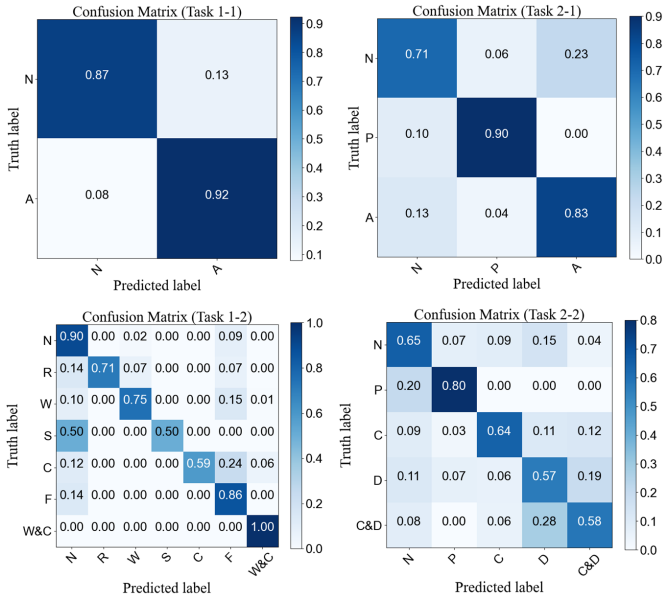

Fig. 8. Confusion matrix of four classification tasks using SPRSound testset.

The algorithm performance is evaluated based on the following metrics including Sensitivity, Specificity, Average Score (AS) and Harmonic Score (HS) [21]. The overall score for each task is the mean value of AS and HS.

$$Score = \frac{AS + HS}{2} \tag{8}$$

The overall total score is the weighted sum of each task's score.

$$Total\ Score = 0.2 \times Score_{1-1} + 0.3 \times Score_{1-2} + 0.2 \times Score_{2-1} + 0.3 \times Score_{2-2} \tag{9}$$

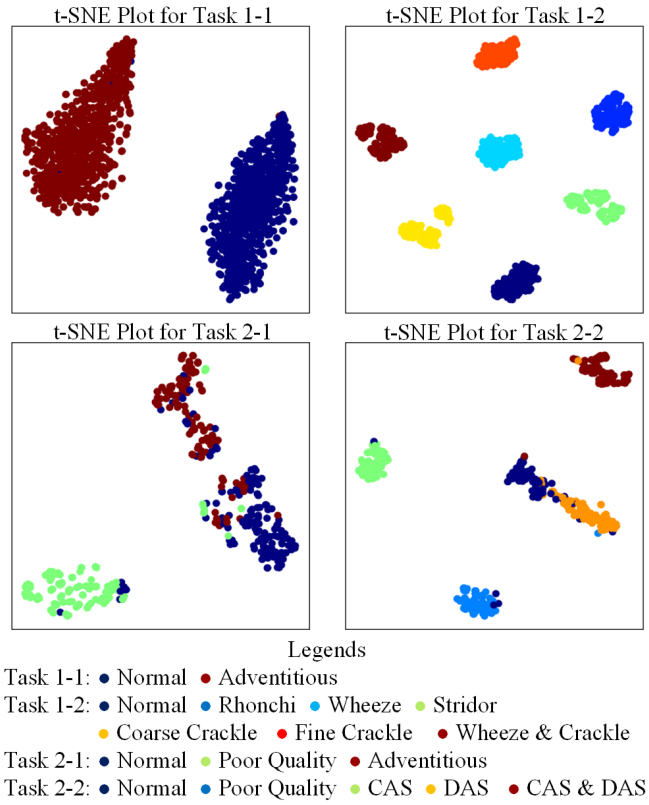

Fig. 9. t-SNE Plot for Task 1-1, Task 1-2, Task 2-1 and Task 2-2.

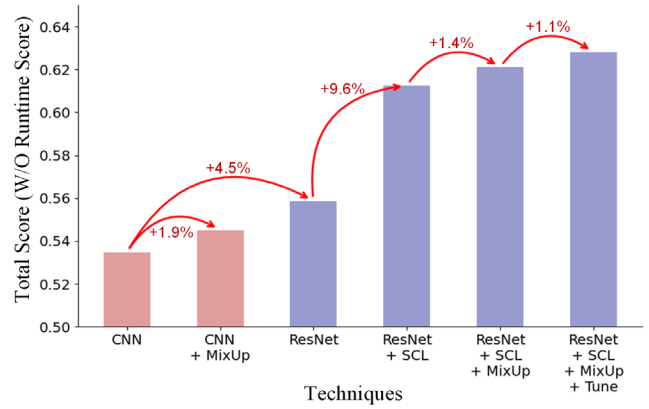

Fig. 10. Classification scores with proposed techniques.

Fig. 10 illustrates the contribution of different techniques to the model performance improvement, based on the results using Grand Challenge'23 testset [35]. It can be observed that SCL brings the most significant classification improvement, MixUp alone is beneficial to both the CNN baseline and the combination of ResNet and SCL.

In addition to classification accuracy, runtime latency is another important feature for real-time respiratory monitoring. Runtime score is defined as the normalized runtime compared to the worst-case baseline reference.

$$Runtime\ Score = 0.1 \times Norm[\log_{10}(t) - \log_{10}(t_0)] \quad (10)$$

where t and t_0 are the runtimes of current model and baseline solution respectively. Fig. 11 compared the total score achieved on SPRSound testset and Grand Challenge'23 testset from

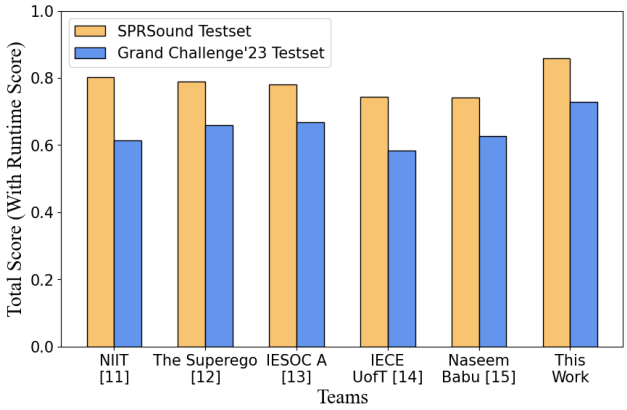

Fig. 11. Comparison of total score achieved on two different testsets (SPRSound testset and Grand Challenge'23 testset).

TABLE II
HARDWARE IMPLEMENTATION PERFORMANCE

Task	Real-time Task 1-1 (3 classes)		Real-time Task 1-2 (8 classes)	
Quantization	FP32	INT8	FP32	INT8
LUT Utilization	153,090 (61.96%)	249,878 (91.17%)	153,090 (61.96%)	249,878 (91.17%)
DSP Utilization	807 (32.02%)	391 (15.52%)	807 (32.02%)	391 (15.52%)
BRAM Utilization	453 (49.67%)	583 (63.93%)	453 (49.67%)	583 (63.93%)
Latency (ms)	61.08	16.18	61.09	16.19
Clock Frequency (MHz)	220	250	220	250
Dynamic Power (mW)	671.5	323.0	671.5	323.0
Throughput (GOPs)	7.25	27.37	7.25	27.35
Energy Efficiency (GOPs/W)	1.50	6.11	1.50	6.11
SPRSound Score	0.855	0.837	0.790	0.775
Grand Challange'23 Score	0.731	0.719	0.601	0.590

different groups. We achieved the highest total score on both testsets and relatively low score variation between the two datasets (refer to Section V.C for more details).

B. Real-time Inference Results

A Xilinx Zynq ZCU102 FPGA is used to implement the real-time respiratory sound inference tasks. As shown in Fig. 12, a laptop controls and monitors the operation of FPGA via UART port, while the audio data is streamed continuously to FPGA via Ethernet port. The classification result and frame latency are transferred back to the laptop for display.

Both FP32 and INT8 models can be deployed in FPGA for real-time Task1-1 and Task1-2. The measured performances are summarized in Table II. It can be observed that, for a fixed quantization scheme (FP32 or INT8), there is no notable difference in hardware utilization between the two tasks. This is because the major difference between these two models is the size of the linear classifier layer, which is only a small portion of the overall model. Hence, the hardware utilization and latency are similar.

Compared to FP32 model, the higher utilization of LUT and BRAM in INT8 model is because of the reduced utilization of DSP slice. In FP32 model, MAC operation is implemented in

TABLE III
BENCHMARK TABLE

	TBioCAS 2022 [21]	BioCAS 2022 [11]	BioCAS 2022 [12]	BioCAS 2022 [13]	BioCAS 2022 [14]	BioCAS 2022 [15]	BioCAS 2023 [16]	BioCAS 2023 [17]	**This Work**	
Feature Extraction	MFCC	STFT	Feature Polymerized	Spectrogram	STFT	MFCC	Wavelet Transform	Spectrogram	MFCC	
Network Model	Scikit-learn	ResNet	AutoGluon	DenseNet	ResNet	2D CNN	Inception Residual	CNN	ResNet	
Model Parameters	NA	12.2M	NA	14.1M	11.5M	0.8M	>12M	0.3M	11.5M	
Representation Learning	NA	Focal Loss	Ensemble Learning	NA	NA	NA	KL-loss	Dual input model	Supervised Contrastive Learning	
Testset	SPRSound	SPRSound	SPRSound	SPRSound	SPRSound	SPRSound	Grand Challenge'23	Grand Challenge'23	SPRSound	Grand Challenge'23
$Score_{1\text{-}1}$	0.7522	0.8886	0.8196	0.8491	0.8926	0.8356	0.8097	0.7560	0.8946	0.7693
$Score_{1\text{-}2}$	0.6157	0.8203	0.7425	0.7473	0.7970	0.7335	0.6666	0.4666	0.8403	0.6318
$Score_{2\text{-}1}$	0.5671	0.7179	0.7114	0.7013	0.7151	0.6696	0.7443	0.6581	0.7635	0.6615
$Score_{2\text{-}2}$	0.3784	0.5331	0.5314	0.5285	0.4543	0.5176	0.6079	0.4583	0.6293	0.5121
Total Score w/o Runtime	0.5621	0.7273	0.6884	0.6928	0.6969	0.6764	0.6932	0.5603	0.7725	0.6293
Runtime (s)	NA	634	216	443	1456	793	7902	2623	165**	201
Total Score w/ Runtime	NA	0.7919	0.7884	0.7692	0.7342	0.7336	0.6931	0.5903	0.8725	0.7294

*Runtime reported by the grand challenge organizer. **Scaled based on the measurement result using local machine.

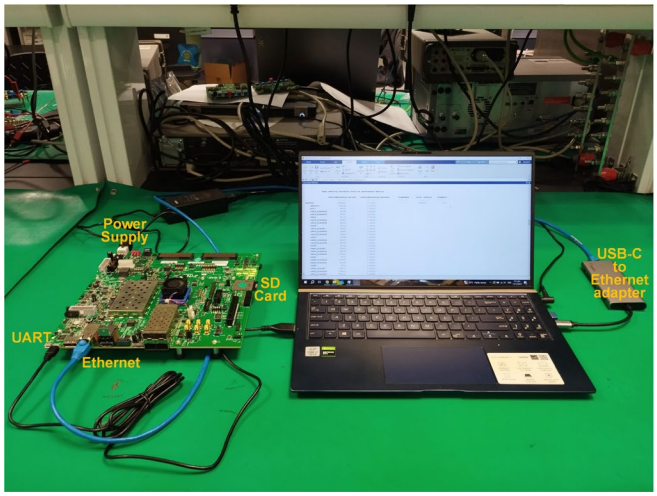

Fig. 12. Xilinx Zynq ZCU102 experiment setup.

DSP slices since it has embedded floating-point multipliers and multiplexers which are suitable to implement FP32 operations. In INT8 model, instead of using DSP slice, MAC operation is implemented only with LUT and BRAM in a more energy efficient way. We want to highlight that the benefit of saving DSP slice outweighs the higher utilization of LUT and BRAM. This can also be reflected from the lower dynamic power consumption and latency of INT8 model.

The FPGA power consumption is measured using AMD-Xilinx System Controller software. The static power is 4.154W, and the dynamic power is 671.5 mW and 323.0 mW for FP32 and INT8, respectively. The clock frequencies for FP32 and INT8 are 220 MHz and 250 MHz, and the throughputs are 7.25 GOPs and 27.37 GOPs (27.35 GOPs for Real-time Task 1-2), respectively. The corresponding energy efficiency is calculated with the ratio of throughput and total power of respective mode.

Compared to a customized accelerator chip, the major limitation of FPGA implementation using existing IP library is the lack of signal flow optimization. For example, the current MATLAB Deep Learning IP does not support pipeline dataflow, resulting in relatively longer processing latency. Nevertheless, FPGA implementation using existing IPs provides a viable approach for rapid validation of algorithm implementation in hardware, paving the way for next step hardware implementation in chip.

C. Benchmarking and Discussion

A summary of the proposed algorithm perfomrnace is shown in Table III and compared with other recent state-of-the-art works reported in Grand Challenge'22 and Grand Challenge'23. All models are trained with the same SPRSound training dataset. It can be observed that the proposed model achieved the shortest runtime among all reported models in both 2022 [47] and 2023 [35]. We also obtained the highest classification score on SPRSound testset. Although [15] and [17] use less weight parameters than this work, both require a huge dynamic memory to store the inter-layer output, which affects the inference speed.

It can be observed that there is relatively large variation between the achieved total scores using two testing datasets. This can be attributed to the difference in the testset data composition. Almost half of the data in SPRSound testset is intra-patient, which is recorded from the same group of participants in the SPRSound training dataset [35]. In contrast, the Grand Challenge'23 testset was constructed with 95 patients outside of the SPRSound training dataset, hence all data are inter-patient. Since the model is trained with SPRSound training dataset, it is reasonable that the total score using SPRSound testset is higher since there is certain correleation between training and test datasets.

Although the proposed framework shows significant performance improvement, it still shows strong dependency on the availability and quality of labelled data, which affects the model's ability to generalize across different respiratory sounds and conditions. To address this limitation, there are two major directions we want to pursue. Firstly, model training with other existing respiratory sound datasets such as ICBHI and HF_Lung_V1 [48] should be explored to further enhance the model generalizability and robustness. Secondly, unsupervised or semi-supervised learning will be investigated to leverage the unlabelled data more effectively.

VI. Conclusion

In this paper, a SCL-based network training framework for respiratory sound classification is presented. The accuracy and robustness of classifier using limited and imbalanced dataset is enhanced through holistic combination of techniques such as data augmentation, SCL, and MixUp. The proposed framework achieved 0.8725 total score (with runtime score) for a ResNet-18 model in both event and record multi-class classification tasks using the SPRSound dataset. The ResNet model is implemented on Xilinx ZCU102 FPGA for real-time respiratory sound classification through a sliding-window approach. The hardware implementation achieves a 16ms latency with less than 2% inference score degradation compared to the ideal software model.

References

[1] "The top 10 causes of death." WHO. Available: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death.

[2] R. X. A. Pramono, S. Bowyer, and E. Rodriguez-Villegas, "Automatic adventitious respiratory sound analysis: A systematic review," *PLOS ONE*, vol. 12, no. 5, pp. 1–43, May 2017.

[3] H. Kiyokawa *et al.*, "Auditory detection of simulated crackles in breath sounds," *Chest*, vol. 119, no. 6, pp. 1886–1892, Jun. 2001.

[4] I. Sen *et al.*, "A comparison of SVM and GMM-based classifier configurations for diagnostic classification of pulmonary sounds," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 7, pp. 1768–1776, Jul. 2015.

[5] N. Jakovljević and T. Lončar-Turukalo, "Hidden markov model based respiratory sound classification," in *Springer Precision Medicine Powered by Health and Connected Health*, pp. 39–43, 2018.

[6] S. Ulukaya *et al.*, "Feature extraction using time-frequency analysis for monophonic-polyphonic wheeze discrimination," in *IEEE Int. Conf. Eng. Med. Bio. Soc. (EMBC)*, Aug. 2015, pp. 5412–5415.

[7] L. Mendes *et al.*, "Detection of crackle events using a multi-feature approach," in *IEEE Int. Conf. Eng. Med. Bio. Soc. (EMBC)*, Aug. 2016, pp. 3679–3683.

[8] J. Acharya and A. Basu, "Deep neural network for respiratory sound classification in wearable devices enabled by patient specific model tuning," *IEEE Trans. Biomed. Circuits Syst.*, vol. 14, no. 3, pp. 535–544, Jun. 2020.

[9] 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 *Interspeech*, Oct. 2020, pp. 2902–2906.

[10] S. B. Shuvo *et al.*, "A lightweight CNN model for detecting respiratory diseases from lung auscultation sounds using EMD-CWT-based hybrid scalogram," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 7, pp. 2595–2603, Jul. 2021.

[11] J. Li *et al.*, "Improving the ResNet-based respiratory sound classification systems with focal loss," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2022, pp. 223–227.

[12] L. Zhang *et al.*, "A feature polymerized based two-level ensemble model for respiratory sound classification," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2022, pp. 238–242.

[13] W.-B. Ma *et al.*, "An effective lung sound classification system for respiratory disease diagnosis using DenseNet CNN model with sound pre-processing engine," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Taipei, Taiwan: IEEE, Oct. 2022, pp. 218–222.

[14] Z. Chen *et al.*, "Classify respiratory abnormality in lung sounds using STFT and a fine-tuned ResNet18 network," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2022, pp. 233–237.

[15] N. Babu *et al.*, "Multiclass categorisation of respiratory sound signals using neural network," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2022, pp. 228–232.

[16] D. Ngo *et al.*, "A deep learning architecture with spatio-temporal focusing for detecting respiratory anomalies," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2023, pp. 1–5.

[17] D. Pessoa *et al.*, "Pediatric respiratory sound classification using a dual input deep learning architecture," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2023, pp. 1–5.

[18] J. Li *et al.*, "LungAttn: advanced lung sound classification using attention mechanism with dual TQWT and triple STFT spectrogram," *Physiol. Meas.*, vol. 42, no. 10, Oct. 2021, Art. no. 105006.

[19] H. Zhu *et al.*, "Automatic pulmonary auscultation grading diagnosis of Coronavirus Disease 2019 in China with artificial intelligence algorithms: A cohort study," *Comput. Methods Programs Biomed.*, vol. 213, p. 106500, Jan. 2022.

[20] B. Liu *et al.*, "Energy-efficient intelligent pulmonary auscultation for post COVID-19 era wearable monitoring enabled by two-stage hybrid neural network," in *IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2022, pp. 2220–2224.

[21] Q. Zhang *et al.*, "SPRSound: open-source SJTU paediatric respiratory sound database," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 5, pp. 867–881, Oct. 2022.

[22] A. Roy and U. Satija, "RDLINet: A novel lightweight inception network for respiratory disease classification using lung sounds," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–13, 2023.

[23] S. Sharmin *et al.*, "A comprehensive analysis on adversarial robustness of spiking neural networks," in *IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2019, pp. 1–8.

[24] S. Yang, H. Wang, and B. Chen, "SIBoLS: robust and energy-efficient learning for spike-based machine intelligence in information bottleneck framework," *IEEE Trans. Cogn. Dev. Syst.*, pp. 1–13, 2024.

[25] B. Deng *et al.*, "Reconstruction of a fully paralleled auditory spiking neural network and FPGA implementation," *IEEE Trans. Biomed. Circuits Syst.*, vol. 15, no. 6, pp. 1320–1331, Dec. 2021.

[26] S. Yang *et al.*, "NADOL: neuromorphic architecture for spike-driven online learning by dendrites," *IEEE Trans. Biomed. Circuits Syst.*, vol. 18, no. 1, pp. 186–199, Feb. 2024.

[27] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *ACM Int. Conf. Mach. Learn. (ICML)*, Jun. 2020.

[28] X. Liu *et al.*, "Self-supervised learning: generative or contrastive," *IEEE Trans KnowL Data Eng*, vol. 35, no. 1, pp. 857–876, Jan. 2023.

[29] E. Fonseca *et al.*, "Unsupervised contrastive learning of sound event representations," in *IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 371–375.

[30] I. Moummad and N. Farrugia, "Pretraining respiratory sound representations using metadata and contrastive learning," in *IEEE Workshop on Appl. of Signal Process. to Audio and Acoust. (WASPAA)*, Oct. 2023, pp. 1–5.

[31] P. Khosla *et al.*, "Supervised contrastive learning," in *Neural Inf. Process. Syst. (NIPS)*, 2020, pp. 18661–18673.

[32] X. Xu *et al.*, "A 2.67GΩ 454nVrms 14.9μW dry-electrode enabled ECG-on-chip with arrhythmia detection," in *IEEE Cust. Integr. Circuits Conf. (CICC)*, Apr. 2023, pp. 1–2.

[33] J. Liu *et al.*, "An ultra-low power reconfigurable biomedical AI processor with adaptive learning for versatile wearable intelligent health monitoring," *IEEE Trans. Biomed. Circuits Syst.*, pp. 1–16, 2023.

[34] P. Peng *et al.*, "Design of an efficient cnn-based cough detection system on lightweight FPGA," *IEEE Trans. Biomed. Circuits Syst.*, vol. 17, no. 1, pp. 116–128, Feb. 2023.

[35] Q. Zhang *et al.*, "Grand challenge on respiratory sound classification for sprsound dataset," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2023.

[36] 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 *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2023, pp. 1–5.

[37] D. Bonet-Solà and R. M. Alsina-Pagès, "A comparative survey of feature extraction and machine learning methods in diverse acoustic environments," *Sensors*, vol. 21, no. 4, p. 1274, Feb. 2021.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, Jun. 2016, pp. 770–778.
[39] H. Zhang *et al.*, "mixup: Beyond empirical risk minimization," in *Int. Conf. Learn. Represent. (ICLR)*, Feb. 2018, pp. 1–13.
[40] D. S. Park *et al.*, "SpecAugment: A simple data augmentation method for automatic speech recognition," in *Interspeech*, Sep. 2019, pp. 2613–2617.
[41] P. Moritz *et al.*, "Ray: A distributed framework for emerging AI applications," in *USENIX Symp. Oper. Syst. Des. Implement. (OSDI)*, 2018, pp. 561–577.
[42] A. Gholami *et al.*, "A survey of quantization methods for efficient neural network inference." arXiv, Jun. 21, 2021.
[43] J. Hu, W. L. Goh, and Y. Gao, "Classification of ECG anomaly with dynamically-biased LSTM for continuous cardiac monitoring," in *IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2023, pp. 1–5.
[44] J. Hu, C. S. Leow, W. L. Goh, and Y. Gao, "Energy efficient software-hardware co-design of quantized recurrent convolutional neural network for continuous cardiac monitoring," in *IEEE Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, Jun. 2023, pp. 1–5.
[45] "FPGA, ASIC, and SoC development—MATLAB & Simulink solutions." Mathworks. Accessed: May 19, 2024. [Online]. Available: https://www.mathworks.com/solutions/ fpga-asic-soc-development.html
[46] Mathworks, "Deep learning processor IP Core—MATLAB & Simulink." Accessed: May 19, 2024. [Online]. Available: https://www.mathworks.com/help/deep-learning-hdl/ug/deep-learning-processor-ip-core.html
[47] Q. Zhang *et al.*, "Grand challenge on respiratory sound classification for SPRSound dataset," in *IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Taipei, Taiwan: IEEE, Oct. 2022, pp. 213–217.
[48] F.-S. Hsu *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, Jul. 2021.

Jinhai Hu (Graduate Student Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from the Nanyang Technological University (NTU), Singapore, in 2021. He is working toward his Ph.D. degree at NTU in collaboration with the Institute of Microelectronics (IME), Agency for Science, Technology and Research (A*STAR), Singapore. His current research interests include software-hardware co-design for biomedical signal analysis and human-machine interface.

J. Hu has received IEEE Circuit and System Society (CASS) 2024 Pre-Doctoral Grant.

Cong Sheng Leow (Graduate Student Member, IEEE) received the B.Eng. degree in electrical and electronic engineering from the Nanyang Technological University (NTU), Singapore, in 2022. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of Michigan, Ann Arbor, MI, USA.

From 2022 to 2023, he was a Research Engineer with the Institute of Microelectronics (IME), Agency for Science, Technology, and Research (A*STAR), Singapore, His current research interests include mixed-signal systems, emerging computing paradigms, and biomedical applications.

Shuailin Tao (Graduate Student Member, IEEE) received the B.Eng. degree in mechanical and aerospace engineering from the Nanyang Technological University (NTU), Singapore, in 2021. He is working toward his Ph.D. degree at NTU in collaboration with the Institute of Microelectronics (IME), Agency for Science, Technology and Research (A*STAR), Singapore. His current research interests include AI algorithm development and model optimization on biomedical signals processing.

Wang Ling Goh (Senior Member, IEEE) received both her Bachelor of Engineering Degree in Electrical and Electronic Engineering and Doctor of Philosophy in Microelectronics from the Department of Electrical and Electronic Engineering at the Queen's University of Belfast in United Kingdom in 1990 and 1995, respectively.

Dr Goh joined the School of Electrical and Electronic Engineering (EEE) as a lecturer and became an Associate Professor in 2004. Prior to her current appointment as Associate Dean (Academic) at the Graduate College, she had held many academic positions such as Deputy Director (Undergraduate) of the Renaissance Engineering Programme (Jun 2019 – Jul 2021), Coordinator for the Final Year Projects at School of EEE (Sep 2018 – Jul 2020), Programme Coordinator of B.Eng. (EEE) (Jun 2014 – May 2017), Member of NTU Teaching Council (Oct 2012 – Jun 2013), Associate Dean (Outreach & External Relations) at the College of Engineering (Jan 2010 – Dec 2012), Assistant Chair of Students (Jul 2008 – Dec 2009) and Assistant Head of Division at School of EEE (Sep 2006 – Jun 2008).

Dr Goh participates actively as General Chair or Advisory/Technical Committee Member in various international conferences. She was the General Conference Chair of the 2016 International Symposium on Integrated Circuits (ISIC 2016), held in Singapore, from December 12-14 in 2016. Dr Goh's research interests include digital/mixed-signal Integrated Circuit (IC), and biomedical and neuromorphic circuits.

Yuan Gao (Member, IEEE) received the B.E and M.E degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2000 and 2002, respectively, and the Ph.D. degree in electrical engineering from the National University of Singapore, Singapore, in 2008.

Since 2007, he has been with the Institute of Microelectronics (IME), Agency for Science, Technology and Research (A*STAR), Singapore. He is currently a principal investigator and principal scientist in the Integrated Circuit Design and Systems (ICDS) Department, where he is leading the next generation intelligent sensor interface IC development. He has authored or coauthored 3 book chapters, more than 120 peer-reviewed international journal and conference papers and has more than 10 US patents granted or filed. He has co-supervised 8 PhD students and he is an accredited A*STAR PhD Scholar supervisor. He received A*STAR Graduate Academy Star Mentor Award in 2023 and IEEE Solid State Circuit Society Outstanding Reviewer Award in 2023. His primary research areas include energy efficient analog and mixed-signal IC design in the emerging areas such as intelligent sensor interface, AI hardware, biomedical microsystem and energy harvesting.

Dr. Gao was TPC member of the IEEE International Solid-State Circuits Conference (ISSCC) between 2015 – 2020 and served as Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS−I: REGULAR PAPERS between 2020 – 2022. Currently he is an Associate Editor of the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS.