# Lightweight Zero-Shot Keyword Spotting via Multi-Granular Knowledge Distillation

Yun-Ting Sun*, Lo-Ya Li†, Tien-Hong Lo*, Jeih-Weih Hung‡, Shih-Chieh Huang§, Berlin Chen*†
*Department of Computer Science and Information Engineering, National Taiwan Normal University, Taiwan
E-mail: {61347015s, teinhonglo, berlin}@ntnu.edu.tw
†Institute of AI Interdisciplinary Applied Technology, National Taiwan Normal University, Taiwan
E-mail: 612k0010c@ntnu.edu.tw
‡Department of Electrical Engineering, National Chi Nan University, Taiwan
E-mail: jwhung@ncnu.edu.tw
§Realtek Semiconductor Corp., Taiwan
E-mail: eric.sc.huang@realtek.com

***Abstract*—Zero-shot keyword spotting (ZSKWS) allows voice interfaces to recognize user-defined keywords without enrollment audio; however, deploying such models on embedded hardware is challenging because memory and compute budgets are severely limited. In this paper, we address this constraint through two complementary countermeasures. First, we compress the acoustic front end with LiCo-Net—a compact encoder built on depthwise-separable convolutions and linear activations that reduces the model footprint to under 0.5 MB while delivering efficient performance. Second, we transfer knowledge from a high-capacity teacher model to the lightweight student one through the synergy of multi-level distillation losses: (i) mean squared error (MSE) in utterance-level logits, (ii) Kullback-Leibler (KL) divergence in phoneme-level probabilities, and (iii) Noise Contrastive Estimation (InfoNCE) loss applied to intermediate features. Notably, with the MSE+KL combination, the resulting lightweight ZSKWS model reduces the equal error rate (EER) of the Qualcomm dataset from $16.8\,\%$ to $10.7\,\%$ (a $36\,\%$ relative reduction), demonstrating the practicality of the knowledge distillation-empowered ZSKWS model even under stringent resource constraints.**

***Index Terms*—Knowledge distillation, zero-shot keyword spotting.**

## I. INTRODUCTION

Keyword spotting (KWS) has progressed from fixed-vocabulary wake-word or keyword detection to few-shot learning [1]–[4] and zero-shot settings [5], where users can register arbitrary phrases in any language for on-device recognition [6]. Along this trajectory, zero-shot KWS (ZSKWS) has become a focus of much current research, aiming to recognize previously unseen keywords without keyword-specific speech.

A leading ZSKWS paradigm is cross-modal contrastive distillation (CMCD) [7], which projects speech utterances and text queries into a shared embedding space, thereby removing the need for keyword-specific audio. Subsequent studies have incorporated phonetic guidance, adaptive contrastive objectives, and extensive data augmentation to improve the accuracy and generalization of the retrieval [8]–[10]. The efficacy of cross-modal methods is largely determined by the audio encoder: richer acoustic representations yield more precise audio–text alignment, which in turn improves the accuracy of keyword retrieval.

To obtain richer acoustic representations, recent work has employed large self-supervised encoders such as Tiny Conformer [11]–[13] and Whisper [14]. While these models achieve state-of-the-art accuracy, their millions of parameters and considerable memory footprint render them unsuitable for deployment on resource-constrained embedded platforms, where memory requirements alone can exceed the total available storage on many IoT devices.

To reduce model size, we adopt the Linearized Convolution Network (LiCo-Net) [15] as the audio encoder. By leveraging depthwise-separable convolutions and linear activations, LiCo-Net reduces the audio encoder to a sub-megabyte footprint, offering efficient detection performance in embedded scenarios. To bridge the remaining accuracy gap against large self-supervised encoders, we employ knowledge distillation (KD) [16], [17], transferring rich intermediate and output representations from a high-capacity teacher to the lightweight student. This KD step markedly improves performance without incurring extra memory or computation costs.

For the teacher network, we incorporate a pre-trained speech encoder [18] as a crucial—though not exclusive—component of the audio-processing pipeline, thereby providing rich, well-calibrated representations across multiple abstraction levels. Knowledge is transferred to the lightweight student through a hierarchy of three cumulative objectives that supervise progressively finer granularity. At the utterance level, a mean-squared-error (MSE) loss aligns posterior logits, ensuring global agreement in keyword probabilities. At the phoneme level, a Kullback–Leibler (KL) divergence refines confidence for individual subword predictions. Finally, at the feature level, an InfoNCE contrastive loss draws the intermediate embeddings toward those of the teacher, enforcing representation consistency throughout the network. This multi-level distillation strategy enables the compact model to inherit both high-level semantics and low-level acoustic detail without increasing its parameter count or computational cost.

979-8-3315-7206-8/25/$31.00  

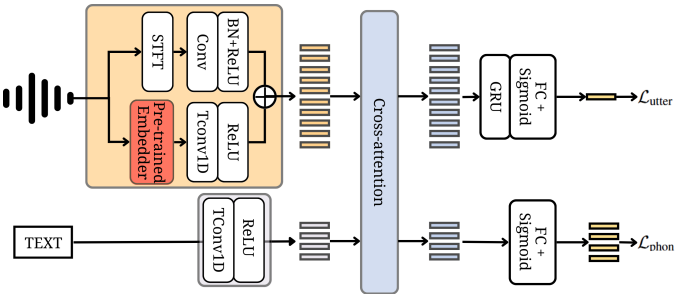


Fig. 1. Architecture of the high-capacity teacher network used for knowledge distillation.

All model variants are trained under the same data regime and evaluated in terms of detection accuracy and memory footprint. Our results show how representation-level distillation choices shift the accuracy–cost frontier for zero-shot KWS. The overall teacher–student architecture and distillation levels are depicted in Figs. 1–3, while inference-time threshold behaviors are summarized in Fig. 4. The contributions are as follows.

- **Compact zero-shot KWS architecture**: We develop a computationally efficient framework that enables real-time inference in resource-constrained scenarios.
- **Knowledge distillation analysis**: We conduct a systematic comparative study of different knowledge distillation objectives for transferring knowledge from a high-capacity teacher to a lightweight student model.
- **Substantial performance gains**: Under resource constraints of 500 kB and 50 mega Floating-Point Operations (FLOPs), the distilled student model reduces EER from 16.8% to 10.7% on the Qualcomm dataset, achieving a 36% relative improvement.

## II. METHOD

The key innovation is multi-granular distillation that transfers knowledge at complementary abstraction levels while preserving both semantic and acoustic cues. To address the performance gap between compact models and large-scale encoders in zero-shot keyword spotting (KWS), we introduce a knowledge distillation framework. Our methodology leverages a high-capacity teacher model based on the Guided Attention Contrastive Learning (GACL) framework to provide comprehensive supervision. A resource-efficient student model based on the Linearized Convolution Network (LiCo-Net) is trained to emulate these rich representational patterns while maintaining computational efficiency.

### A. Teacher Model

As shown in Fig. 1, a high-capacity teacher based on the GACL framework provides rich supervision signals for zero-shot KWS.

**Audio Encoder.** The encoder processes the waveform through two parallel paths. In the trainable path, raw audio is first converted into 40-dimensional log-mel filter bank frames with a 25 ms window and 10 ms hop; these frames then pass through two one-dimensional convolutions with kernel size 3, the first using stride 2 and the second stride 1, each followed by batch normalization and ReLU to capture local acoustic detail. In the semantic path, a frozen Google Speech Embedder [18] produces 96-dimensional embeddings every 80 ms from 775 ms segments, providing high-level phonetic cues without introducing additional trainable parameters. The two streams are aligned in temporal resolution and dimensionality using a one-dimensional transposed convolution (kernel size 5, stride 4) and a linear projection, then concatenated and mapped to 128 dimensions, forming the audio embedding matrix $\mathbf{E}^{\text{a}} \in \mathbb{R}^{T_{\text{a}} \times 128}$, where $T_{\text{a}}$ denotes the number of 20 ms time steps.

**Text Encoder.** A frozen Grapheme-to-Phoneme model [19] maps character sequences to phonemes; its final hidden states pass through a linear layer with ReLU to 128 dim and receive sinusoidal positional encoding, yielding $\mathbf{E}^{\text{t}} \in \mathbb{R}^{T_{\text{t}} \times 128}$ with $T_{\text{t}}$ as the phoneme-sequence length.

**Pattern Extractor.** The pattern extractor adopts a multi-layer cross-attention block that aligns speech and text by using the phoneme embedding sequence as the *query* and the fused acoustic representation as both the *key* and the *value*. Let $Q \in \mathbb{R}^{T_{\text{t}} \times d}$ be the phoneme-embedding matrix produced by the text encoder, and let $K, V \in \mathbb{R}^{T_{\text{a}} \times d}$ be the time-aligned acoustic descriptors and their associated content vectors generated by the audio branch. Cross-attention is computed as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right) V, \tag{1}$$

yielding a sequence of joint embedding vectors, denoted as $\mathbf{E}^{\text{j}} \in \mathbb{R}^{T_{\text{t}} \times 128}$, which serve as pattern tokens whose activations highlight the acoustic segments that best correspond to each phoneme.

**Pattern Discriminator.** The joint embeddings $\mathbf{E}^{\text{j}}$ are processed by a GRU network and fed to dual classification heads. To detect utterance-level matching between audio and keywords, a GRU layer followed by a fully connected layer is applied:

$$\mathbf{z}_u = \mathbf{W}_u \cdot \text{GRU}(\mathbf{E}^{\text{j}}) + b_{\text{u}} \in \mathbb{R}^{1 \times 1}. \tag{2}$$

To capture phoneme-level alignment, temporal segments are extracted or transformed from $\mathbf{E}^{\text{j}}$, denoted as $(\mathbf{E}^{\text{j}})_{\text{temp}}$, and then passed through a separate classification head:

$$\mathbf{z}_p = \mathbf{W}_p \cdot (\mathbf{E}^{\text{j}})_{\text{temp}} + b_{\text{p}} \in \mathbb{R}^{T_{\text{t}} \times 1}. \tag{3}$$

The final matching probabilities are obtained by applying a sigmoid activation to these logits, scaled by temperature $\tau$.

Instead of using explicit contrastive losses, a CTC loss [20] is applied to guide alignment and a sigmoid bound loss is introduced to improve detection stability and reduce false alarms.

### B. Student Model

As shown in Fig. 2, the student model retains the teacher's cross-attention pattern extractor and GRU discriminator, while

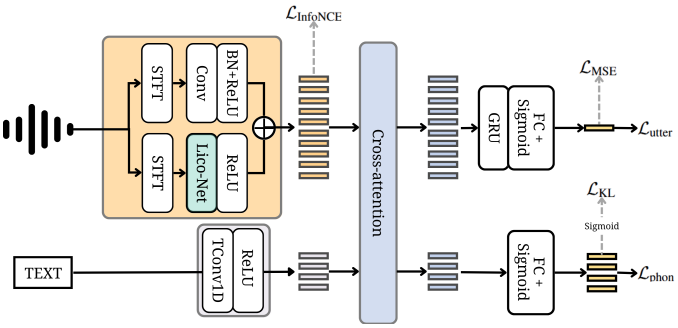


Fig. 2. Student model pipeline. The LiCo-Net branch highlighted in green replaces the original audio front end, and its output is concatenated with the remaining feature stream before cross-attention. The three distillation losses—(i) utterance-level MSE, (ii) phoneme-level KL, and (iii) feature-level InfoNCE—are applied.

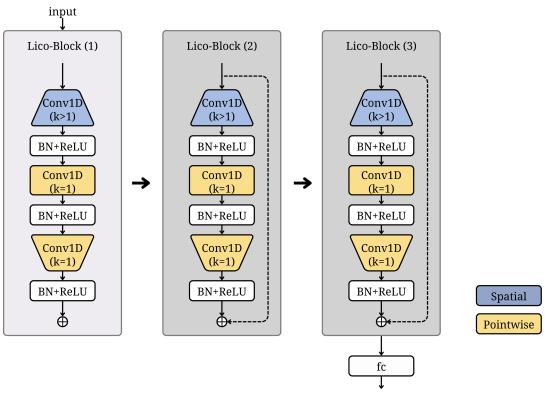


Fig. 3. LiCo-Net audio encoder consisting of three depthwise separable convolution blocks with linearized activations.

replacing the heavy audio front end with a dual-path encoder optimized for edge devices.

**Dual-Path Audio Encoder.** The student model incorporates a novel dual-path audio encoder, representing the primary architectural divergence from the teacher model. This component replaces the teacher encoder while rigorously maintaining computational efficiency, a critical prerequisite for on-device deployment.

The first path of this encoder mirrors the trainable path described in the teacher model audio encoder. The second path processes the same audio input, but utilizes a LiCo-Net encoder. This path operates on frames of 400 samples with a 160-sample hop, producing 32 Mel bins that span 60 to 3800 Hz, a configuration consistent with that used by the Google Speech Embedder. The internal architecture of the LiCo-Net encoder, which is configured with an input dimension of 32, a kernel size of 5, three linearized depthwise separable convolution blocks, and an expansion factor of 6, is illustrated in Fig. 3. The entire LiCo-Net branch maintains a compact footprint, occupying less than 250 kB. Feature maps from both branches are then concatenated, positionally encoded, and fed into the shared cross-attention module. By preserving the teacher model's downstream topology while replacing its computationally intensive audio front-end, the student model maintains a memory footprint below 500 kB.

**Mean Squared Error (MSE) KD Loss.** At the utterance level, we align teacher and student logits $\mathbf{z}^t_{u,i}, \mathbf{z}^s_{u,i}$ using

$$\mathcal{L}_{\text{MSE}} = \tau^2 \frac{1}{N} \sum_{i=1}^{N} \left\| \mathbf{z}^t_{u,i} - \mathbf{z}^s_{u,i} \right\|_2^2, \tag{4}$$

with $\tau = 2$.

**Kullback–Leibler (KL) KD Loss.** For each valid frame $j$ in utterance $i$, logits are converted to probabilities

$$\mathbf{s}^t_{ij} = \sigma\big(\mathbf{z}^t_{p,ij}/\tau\big) \quad \text{and} \quad \mathbf{s}^s_{ij} = \sigma\big(\mathbf{z}^s_{p,ij}/\tau\big).$$

The per-frame KL divergence $\Delta_{ij}$ is defined as:

$$\Delta_{ij} = \mathbf{s}^t_{ij} \log \frac{\mathbf{s}^t_{ij} + \epsilon}{\mathbf{s}^s_{ij} + \epsilon} + \big(1 - \mathbf{s}^t_{ij}\big) \log \frac{1 - \mathbf{s}^t_{ij} + \epsilon}{1 - \mathbf{s}^s_{ij} + \epsilon}, \tag{5}$$

and the phoneme-level distillation loss is

$$\mathcal{L}_{\text{KL}} = \tau^2 \frac{1}{N} \sum_{i=1}^{N} \frac{1}{T_i} \sum_{j=1}^{T_i} \Delta_{ij}, \tag{6}$$

with $\tau = 2$.

**Intermediate-Feature InfoNCE Loss.** Let $\tilde{\mathbf{E}}^s_a \in \mathbb{R}^{N \times d}$ and $\tilde{\mathbf{E}}^t_a \in \mathbb{R}^{N \times d}$ be the $\ell_2$–normalized student and teacher embedding matrices for a minibatch of size $N$. Denote their $i$th rows (the embeddings for sample $i$) by $\tilde{\mathbf{e}}^s_{a,i} \in \mathbb{R}^d$ and $\tilde{\mathbf{e}}^t_{a,i} \in \mathbb{R}^d$, respectively. Then the contrastive objective is

$$\mathcal{L}_{\text{InfoNCE}} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{\exp\big(\tilde{\mathbf{e}}^s_{a,i} \cdot \tilde{\mathbf{e}}^t_{a,i}/\tau\big)}{\sum_{j=1}^{N} \exp\big(\tilde{\mathbf{e}}^s_{a,i} \cdot \tilde{\mathbf{e}}^t_{a,j}/\tau\big)}, \tag{7}$$

where $\tau = 0.2$. This loss aligns each student embedding with its teacher counterpart while pushing it away from other teacher embeddings.

Total loss is defined as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{utter}} + \mathcal{L}_{\text{phon}} + \alpha\, \mathcal{L}_{\text{MSE}} + \beta\, \mathcal{L}_{\text{KL}} + \gamma\, \mathcal{L}_{\text{InfoNCE}},$$

where we set $\alpha = 1$, $\beta = 1$, and $\gamma = 0.1$.

## III. EXPERIMENTS

To validate the effectiveness of the proposed multi–granular KD framework, we conducted experiments on three publicly available corpora and report standard detection metrics alongside model efficiency figures.

### A. Datasets

- **LibriPhrase** [21]: This dataset is exclusively for model training. To enhance robustness, the audio is augmented with MS-SNSD noise (0-20 dB SNR) [22]. The corresponding test sets, derived from train-others-500, are categorized into LibriPhrase Easy (LE) and LibriPhrase Hard (LH) based on complexity.
- **Google Speech Commands V1 (G)** [23]: The evaluation was carried out on 10 short commands from the GSC V1 test dataset. For this assessment, each anchor keyword was paired with all other keywords to form negative examples.

TABLE I
OVERALL PERFORMANCE COMPARISON.

| Method | AUC (%) ↑ | | | | EER (%) ↓ | | | | Complexity | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | G | Q | LE | LH | G | Q | LE | LH | LiCo Params | Total Params | FLOPs |
| Teacher (GACL) | 99.51 | 99.97 | 98.04 | 86.92 | 3.22 | 0.67 | 6.11 | 20.71 | – | 655K | 93 M |
| Student w/o KD | 97.29 | 91.54 | 95.37 | 77.89 | 8.20 | 16.80 | 10.87 | 29.08 | 233K | 464K | 45 M |
| Student (MSE) | 98.35 | 90.76 | **97.60** | **82.38** | 6.31 | 16.27 | **6.82** | **25.03** | | | |
| Student (MSE, KL) | **98.35** | **96.48** | 97.11 | 81.86 | **6.11** | **10.72** | 7.58 | 25.40 | | | |
| Student (MSE, KL, InfoNCE) | 97.83 | 94.12 | 97.32 | 81.66 | 7.00 | 13.33 | 7.36 | 25.94 | | | |

- **Qualcomm Keyword Speech (Q)** [24]: This dataset features four proprietary wake words recorded on mobile devices, including those with acoustically similar initial phonemes that challenge discriminative performance. Lacking explicit negative pairs, non-anchor keywords are treated as negatives during testing.

All audio was resampled to 16 kHz and peak–normalized to $-20$ dB LUFS.

### B. Implementation Details

All models are implemented in TensorFlow and trained for 20 epochs on a single NVIDIA A10 (24 GB). We use the Adam optimizer with an initial learning rate of $1 \times 10^{-3}$ and a global batch size of 512. To ensure stable convergence and prevent gradient explosion, gradients are clipped to an $\ell_2$ norm of 1.0. At evaluation, we report two sample-level metrics: the Equal Error Rate (EER), with thresholds chosen by equalizing false-accept and false-reject rates on the development split, and the Area Under the ROC Curve (AUC).

## IV. RESULTS

### A. Knowledge Distillation Objective Comparison

Table I summarizes the performance of our student model under three distinct distillation objectives, juxtaposed with a high-capacity GACL teacher and a non-distilled baseline, all evaluated under an identical inference budget, with shared preprocessing, identical decoding. The teacher consistently achieves the highest performance, while the baseline student shows substantial degradation, underscoring the critical role of knowledge distillation for calibration and robust generalization.

When applying **utterance-level MSE** in isolation, performance on the LE dataset significantly improved, with AUC rising from 95.37% to 97.60% and EER decreasing from 10.87% to 6.82%, representing a 37% relative error reduction. For LH, this configuration yielded an AUC of 82.38% and an EER of 25.03%. This objective also enhanced Google Speech Commands AUC by an absolute 1.10%, increasing it from 97.29% to 98.35%.

The incorporation of **phoneme-level KL divergence** further bolstered performance, most notably on the Qualcomm subset. Here, AUC increased from 90.76% to 96.48%, and EER decreased from 16.27% to 10.72%. On Google Speech Commands, this combined objective reduced EER to 6.11% while maintaining the 98.35% AUC, thereby demonstrating the advantages of fine-grained logit alignment.

Conversely, integrating the **feature-level InfoNCE loss** enhanced the robustness of intermediate embeddings but led to slightly degraded overall performance compared to MSE+KL configuration. Specifically, on the Google Speech Commands benchmark, InfoNCE produced an AUC of 97.83 % and an EER of 7.00 %, while on the Qualcomm dataset it yielded an AUC of 94.12 % and an EER of 13.33 %.

These results show a modest AUC reduction and a higher EER than MSE+KL. We conjecture that the InfoNCE objective underperforms due to both sub-optimal hyperparameters and the negative-sample sampling scheme. Even after filtering same-keyword pairs, the mini-batch–limited negative pool can include acoustically confusable yet distinct words (e.g., with similar phoneme sequences or prosody). This can introduce false negatives, perturb representation alignment and threshold calibration, and ultimately yield the observed performance degradation. Furthermore, the computational overhead introduced by each distillation objective remains negligible during inference, with all variants maintaining the same 45 MFLOP budget while achieving different accuracy-efficiency trade-offs, indicating that distillation objectives shift accuracy without inflating complexity, allowing practitioners to select a loss mix that best matches deployment targets.

### B. Threshold-Sweeping Analysis on Qualcomm

As shown in Fig. 4, we sweep the posterior-probability threshold $\theta$ that triggers a keyword prediction. The Qualcomm corpus, featuring proprietary wake words and Indian-accented speakers, poses a challenging edge-KWS benchmark. Without distillation, the student model achieves its lowest EER at $\theta = 0.40$, corresponding to a 25% false positive rate (FPR) and peak accuracy of 0.75. Introducing utterance-level MSE shifts the optimal threshold to $\theta = 0.70$, halves the FPR to 12.5%, and raises accuracy to 0.86, indicating better calibrated logits and improved separation between positive and negative pairs. Augmenting with logit-level KL divergence reduces the FPR by over an order of magnitude to 1.06% at the EER threshold and elevates peak accuracy to 0.91 while preserving recall above 0.87. Adding a feature-level InfoNCE term degrades performance, with FPR increasing to 13.55% at $\theta = 0.60$, confirming that MSE+KL offers the best precision–recall trade-off under the 45 MFLOP budget. The threshold analysis reveals that knowledge distillation not only improves overall accuracy but also enhances calibration, yielding more reliable confidence estimates and simpler threshold selection for deployment.

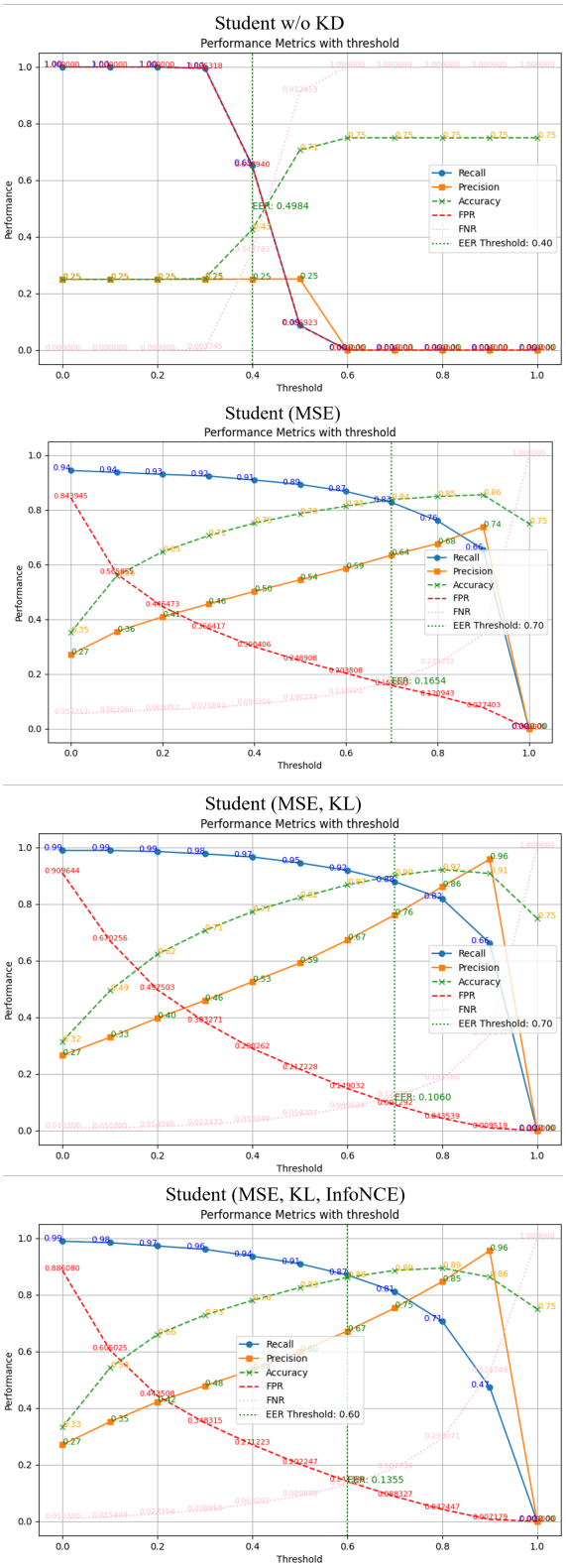


Fig. 4. Inference-time threshold-sweeping curves on the Qualcomm test set. Each subplot reports recall, precision, accuracy, false-positive rate (FPR), false-negative rate (FNR), and the dashed vertical line marking the EER threshold.

## V. CONCLUSION

We have presented a sub-megabyte zero-shot keyword spotting framework that integrates a LiCo-Net encoder with three-level knowledge distillation—spanning utterance, phoneme, and feature representations. Operating under a 500 kB/50 MFLOP constraint, our distilled student model achieves a 10.7 % EER on the Qualcomm test set, representing a 36 % relative reduction, while preserving strong performance across other benchmarks. These results demonstrate that accurate zero-shot KWS is feasible even under severe resource limitations. Future work will explore avenues to further boost detection accuracy, enhance resilience to accent and pronunciation variability, and reduce false alarms, thereby improving the robustness and reliability of on-device voice interfaces.

## VI. ACKNOWLEDGMENT

This work was supported in part by Realtek Semiconductor Corporation under Grant Number 113KK01103. Any findings and implications in the paper do not necessarily reflect those of the sponsors.

## REFERENCES

[1] S. M. Kye, Y. Jung, H. B. Lee, S. J. Hwang, and H. Kim, "Meta-learning for short utterance speaker recognition with imbalance length pairs," *arXiv preprint arXiv:2004.02863*, 2020.

[2] A. Parnami and M. Lee, "Few-shot keyword spotting with prototypical networks," in *2022 7th International Conference on Machine Learning Technologies (ICMLT)*, ser. ICMLT 2022, ACM, Mar. 2022, pp. 277–283.

[3] M. Rusci and T. Tuytelaars, "Few-shot open-set learning for on-device customization of keyword spotting systems," in *booktitle = Interspeech 2020.*, Aug. 2023, pp. 2768–2772.

[4] L.-Y. Li, T.-H. Lo, J.-W. Hung, S.-C. Huang, and B. Chen, "Few-shot open-set keyword spotting with multi-stage training," in *2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, IEEE, 2024, pp. 1–5.

[5] C. Cioflan, L. Cavigelli, and L. Benini, "Boosting keyword spotting through on-device learnable user speech characteristics," in *TinyML Research Symposium*, 2024.

[6] P. M. Reuter, C. Rollwage, and B. T. Meyer, "Multilingual query-by-example keyword spotting with metric learning and phoneme-to-embedding mapping," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2023, pp. 1–5.

[7] H. Kim, S. Kim, J. Lee, and S. Yoon, "Learning audio-text agreement for open-vocabulary keyword spotting," *arXiv:2206.15486*, 2022.

[8] Y.-H. Lee and N. Cho, "Phonmatchnet: Phoneme-guided zero-shot keyword spotting for user-defined keywords," *arXiv:2308.16511*, 2023.

[9] A. Zhang, P. Zhou, K. Huang, Y. Zou, M. Liu, and L. Xie, "U2-kws: Unified two-pass open-vocabulary keyword spotting with keyword bias," in *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, IEEE, 2023, pp. 1–8.

[10] A. Navon, A. Shamsian, N. Glazer, G. Hetz, and J. Keshet, "Open-vocabulary keyword-spotting with adaptive instance normalization," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2024, pp. 11 656–11 660.

[11] K. Nishu, M. Cho, P. Dixon, and D. Naik, "Flexible keyword spotting based on homogeneous audio-text embedding," *arXiv:2308.06472v1*, 2023.

[12] Z. Ai, Z. Chen, and S. Xu, "MM-KWS: Multi-modal prompts for multilingual user-defined keyword spotting," in *Proc. Interspeech*, 2024, pp. 2415–2419.

[13] K. Nishu, M. Cho, and D. Naik, "Slick: Exploiting subsequences for length-constrained keyword spotting," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2025, pp. 1–5.

[14] L. Kewei, Z. Hengshun, S. Kai, D. Yusheng, and D. Jun, "Phoneme-level contrastive learning for user-defined keyword spotting with flexible enrollment," *arXiv preprint arXiv:2412.20805*, 2024.

[15] H. Yang, Z. Yang, L. Wan, *et al.*, "Lico-net: Linearized convolution network for hardware-efficient keyword spotting," *arXiv preprint arXiv:2211.04635*, 2022.

[16] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[17] Y. Kim and A. M. Rush, "Sequence-level knowledge distillation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2016, pp. 1317–1327. arXiv: 1606.07947.

[18] J. Lin, K. Kilgour, D. Roblek, and M. Sharifi, "Training keyword spotters with limited and synthesized speech data," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 7474–7478. arXiv: 2002.01322.

[19] K. Park and J. Kim. "*g2pe: Grapheme-to-Phoneme Conversion in Python*." (2019), [Online]. Available: https://github.com/Kyubyong/g2p.

[20] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer, "Sigmoid loss for language image pre-training," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 11 975–11 986.

[21] K. Nishu, M. Cho, and D. Naik, "Matching latent encoding for audio-text based keyword spotting," *arXiv preprint arXiv:2306.05245*, 2023.

[22] C. K. Reddy, E. Beyrami, J. Pool, R. Cutler, S. Srinivasan, and J. Gehrke, "A scalable noisy speech dataset and online subjective test framework," *arXiv preprint arXiv:1909.08050*, 2019.

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

[24] B. Kim, M. Lee, J. Lee, Y. Kim, and K. Hwang, "Query-by-example on-device keyword spotting," *arXiv:1910.05171*, 2020.