EfficientMic: Adaptive Acoustic Sensing with a Single Microphone for Smart Infrastructure

Jack Adiletta
Worcester Polytechnic Institute
Worcester, MA, USA

Khan Mohammad Nur Hossain
Worcester Polytechnic Institute
Worcester, MA, USA

Matthew Reynolds
Columbia University
New York, NY, USA

Shiwei Fang
Augusta University
Augusta, GA, USA

Bashima Islam
Worcester Polytechnic Institute
Worcester, MA, USA

Abstract

Acoustic sensing is essential in smart infrastructure for occupancy detection, appliance monitoring, and anomaly alerts, where latency and continuous operation are critical. However, fixed high-rate sampling drains power and storage in embedded systems. Prior adaptive sampling approaches often rely on narrow-band speech, analog filters, multi-microphone arrays, or task-specific heuristics, each increasing hardware complexity or reducing generalizability. We present EfficientMic, a digital, aliasing-aware framework for adaptive acoustic sensing using only a single microphone. Without analog components, EfficientMic detects aliasing from Short-time Fourier Transform (STFT) features using a lightweight XGBoost model and dynamically adjusts the sampling rate in real time to preserve fidelity across diverse sounds. Deployed on a Raspberry Pi and evaluated on two environmental sound classification datasets, EfficientMic reduces storage and energy use by over 70% and 50%, respectively, with classification accuracy within 1.2% of full-rate baselines. This task-agnostic, hardware-independent design enables scalable, low-power sensing for smart environments.

CCS Concepts

• **Computer systems organization** → **Sensor networks**; Embedded software; • **Computing methodologies** → *Classification and regression trees*; *Neural networks.*

Keywords

adaptive sampling, machine learning, smart infrastructure

ACM Reference Format:
Jack Adiletta, Khan Mohammad Nur Hossain, Matthew Reynolds, Shiwei Fang, and Bashima Islam. 2025. *EfficientMic*: Adaptive Acoustic Sensing with a Single Microphone for Smart Infrastructure. In *The 12th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BUILDSYS '25), November 19–21, 2025, Golden, CO, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3736425.3770108

1 Introduction

Microphones are essential sensing components in smart infrastructure, supporting applications such as occupancy monitoring,

ACM ISBN 979-8-4007-1945-5/25/11
https://doi.org/10.1145/3736425.3770108

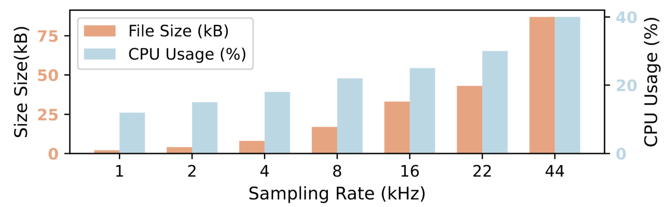

Figure 1: Reducing sampling from 44 kHz to 2 kHz cuts CPU use by 50% and storage by 95% on an ARM Cortex-A8. [5].

appliance activity detection, and acoustic anomaly recognition (e.g., alarms, glass breaking) [2]. These applications require always-on operation, as acoustic events occur unpredictably and must be detected with minimal latency. However, selecting a suitable sampling rate is challenging due to the wide range of signal characteristics in smart environments, which span from low-frequency appliance hums to high-frequency, transient events. A high fixed sampling rate preserves fidelity but imposes significant energy and storage costs on embedded platforms [7]. A low sample rate, while resource-efficient, risks aliasing, a distortion caused by under sampling high-frequency components [4]. Figure 1 shows that reducing the rate from 44 kHz to 2 kHz lowers CPU usage by 50% and storage by 95%, but at the expense of audio quality and downstream performance [1]. Since no single fixed rate performs optimally across all acoustic conditions, adaptive sampling is essential. By increasing the sampling rate only when high-frequency content is present, systems can preserve signal fidelity while conserving energy and storage during low-frequency or silent periods.

Analog filters suppress frequencies above half the sampling rate but require additional circuitry, increasing hardware complexity, cost, and limiting scalability [10–12]. In practice, most commercial IoT and embedded platforms digitize audio immediately after capture and omit analog filtering to simplify design [12]. This leaves only digital streams accessible, reinforcing the need for aliasing mitigation entirely within the digital domain. Multi-microphone systems [5] address aliasing by dedicating at least one microphone to continuous high-rate sampling, but this adds hardware cost, power consumption, and synchronization complexity. Adaptive approaches based on energy-aware models or task-specific neural networks [6, 12] attempt to reduce computational load, yet often rely on heuristics or tightly coupled inference pipelines. These methods do not explicitly detect aliasing and tend to generalize poorly across varied tasks and environments. Naive downsampling without frequency-aware adaptation introduces spectral distortion,

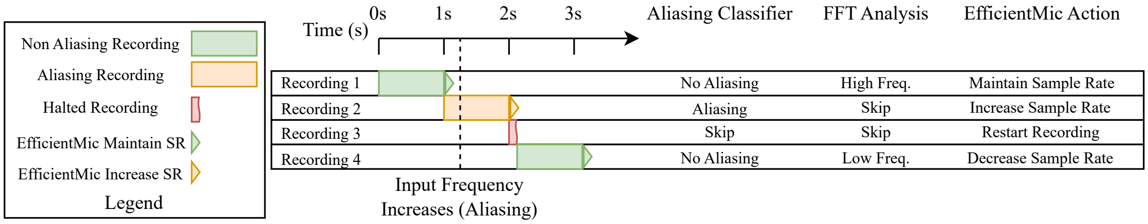

Figure 2: Timeline view of *EfficientMic*'s adaptive response to aliasing across four 1-second recordings. Aliasing in Recording 2 is detected during Recording 3, which is halted. The system increases the sampling rate in Recording 4 to prevent aliasing.

degrading signal quality and downstream performance [1]. Therefore, we need a task-agnostic, real-time adaptive sampling framework that is aliasing-aware, fully digital, hardware-compatible, and scalable for broad applications in environmental acoustic sensing.

To address these limitations, we introduce *EfficientMic*, a lightweight, aliasing-aware adaptive sampling framework that dynamically adjusts microphone sampling rates in real time using a single microphone. *EfficientMic* detects aliasing directly from Short-time Fourier Transform (STFT) features using a compact XGBoost classifier [3] and increases the sampling rate only when necessary to preserve signal fidelity. This approach reduces computational and storage overhead without requiring task-specific tuning or additional hardware. *EfficientMic* is designed for deployment on low-power, always-on embedded systems that lack analog filtering or multi-channel support, making it suitable for a wide range of smart infrastructure applications. Our core contributions are:

- *EfficientMic*, the first aliasing-aware adaptive sampling framework operating fully in the digital domain using only a single microphone.
- A lightweight aliasing detection pipeline using STFT and XGBoost classifier, enabling real-time adaptation on embedded platforms.
- A curated dataset of 587,350 unique 1-second aliased and non-aliased STFT and MFCC audio representations at varying frequencies derived from public datasets to support reproducibility future research available at https://zenodo.org/records/16712803.

2 EfficientMic: Adaptive Sampling Framework

EfficientMic is a lightweight, aliasing-aware adaptive sampling framework that dynamically adjusts microphone sampling rates in real time based on incoming audio characteristics. Designed for always-on, single-microphone systems without analog filtering or multi-channel hardware, *EfficientMic* operates entirely in the digital domain and is optimized for low-power embedded platforms.

As shown in Figure 3, the system begins by sampling the audio stream at a current rate. Each audio segment is analyzed by an ML-based aliasing detector that uses STFT features as input. This module is implemented using an XGBoost classifier trained to distinguish aliased from non-aliased signals. XGBoost is chosen for its low memory footprint, fast inference, and high classification accuracy in resource-constrained settings (see Section 4.2).

If aliasing is detected, the system increases the sampling rate to preserve signal fidelity. If not, the audio is passed to a frequency analysis module, implemented via FFT, to estimate the dominant frequency components in the signal. If the energy is concentrated in low-frequency bands, the sampling rate is decreased to conserve

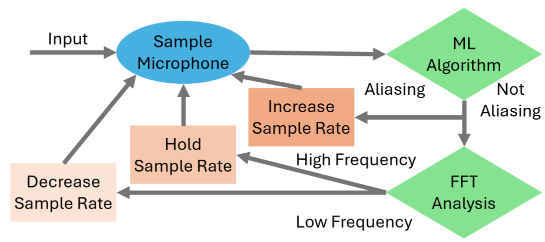

Figure 3: *EfficientMic* Adaptive Sampling Framework

power. If high-frequency energy is present but aliasing is not detected, the system holds the current sampling rate. This dual-stage decision pipeline allows *EfficientMic* to adjust its behavior at runtime based on both aliasing risk and frequency content.

Figure 2 illustrates *EfficientMic*'s behavior over four consecutive recording windows. In the first window, the system detects no aliasing and maintains the current rate. In the second window, high-frequency content triggers aliasing, which is detected in the third. In the fourth window, *EfficientMic* increases the sampling rate to mitigate aliasing. This closed-loop pipeline enables low-latency, frame-level decisions that balance audio fidelity with resource efficiency. By framing aliasing detection as a lightweight classification task and deferring rate decisions to an FFT-based module, *EfficientMic* minimizes power usage while maintaining responsiveness. In stable environments, the system reduces the frequency of adjustments to conserve energy, making it well suited for real-world, always-on acoustic sensing in smart infrastructure.

3 Implementation Details

Aliasing Detection Dataset. To our knowledge, no public dataset exists for training machine learning models to perform dynamic sampling based on aliasing detection. We construct a novel dataset using 97,891 unique 1-second clips from UrbanSound8K [9], ESC-50 [8], and MAVD [13], totaling over 27 hours of diverse, real-world audio. The dataset includes aliased and non-aliased versions of each clip across six target sampling rates (22 kHz, 16 kHz, 8 kHz, 4 kHz, 2 kHz, and 1 kHz), totaling 587,346 pairs of aliasing and non aliasing audio samples across a broad range of environmental sounds. Most sampling rates are balanced with equal numbers of aliased and non-aliased examples, with the exception of 22 kHz where aliasing is rare. Overall, this dataset provides a diverse and carefully curated benchmark for aliasing detection and supports future works on learning-based adaptive sampling frameworks.

We generate an aliased and a non-aliased copies of each audio clip. The non-aliased version is created by applying a sixth-order Butterworth low-pass filter with a cutoff frequency equal to half

WASPAA25/Figures/usage_v3.pdf

Figure 4: Proposed *EfficientMic* achieves similar performance on ESC-50 and UrbanSound8K with 70.3% lower storage and 40.4% lower energy requirement. Energy requirements and storage needed are the same for both datasets for fixed sampling rates due to similar models and data size during inference. This time and storage includes the time and storage overhead of *EfficientMic*.

the target sampling rate (11 kHz, 8 kHz, 4 kHz, 2 kHz, 1 kHz or 500 Hz), followed by resampling at the target frequency which is twice the cutoff (22 kHz, 16 kHz, 8 kHz, 4 kHz, 2 kHz, or 1 kHz) to satisfy the Nyquist criterion [10]. The aliased version is generated by resampling the unfiltered signal at the same target rates, resulting in spectral folding of high-frequency components. To ensure that aliased and non-aliased pairs are meaningfully distinct, we discard samples where energy above the cutoff frequency is less than 10% of the total spectral energy.

Model Deployment and Evaluation Details. We deploy the *EfficientMic* framework on a Raspberry Pi and evaluate its performance in terms of classification accuracy, F1 score, latency, and energy consumption. Latency and energy are measured for two components: STFT computation and model inference. Latency is recorded using

Python timers, and power consumption is measured with an Agilent 34450A Digital Multimeter by recording voltage and average current during execution. Total energy consumption is computed as the product of execution time and power. We exclude the latency and energy required for audio recording, as all clips are 1 second long and sampling-related overhead is consistent across configurations. For a robust evaluation across all sample rates, we perform K-fold cross-validation using the built-in partitioning structure of each dataset: 10 folds for UrbanSound8K and 5 folds for ESC-50. Final performance metrics, including accuracy and F1 score, are averaged across all folds to assess *EfficientMic*'s aliasing detection accuracy and downstream task utility.

To demonstrate the practical benefits of *EfficientMic*, we evaluate its impact on sound event detection. Sound event detection is chosen as a representative task due to its frequency diversity and temporal variability, making it generalizable to use cases such as keyword spotting or noise suppression. We evaluate two compact models suitable for embedded deployment: (1) a convolutional neural network (CNN) with two convolutional layers, two fully connected layers, max pooling, batch normalization, and dropout; and (2) A multi-layer LSTM that encodes sequential input using the final hidden state of the last layer to perform classification via a fully connected layer. These lightweight models provide consistent baselines for comparing performance across sampling strategies.

4 Results and Discussion

Effects of Dynamic Sampling Rate on Acoustic Event Detection Performance. Figure 4 shows how sampling rate impacts classification accuracy (CNN), storage (recorded .wav size), and STFT energy cost. While 44 kHz sampling yields the highest classification accuracy (Figure 4a), it significantly increases storage (Figure 4b) and STFT energy cost (Figure 4c). Lowering the rate to 2 kHz reduces storage by 95.29% and energy by 80.13%, but introduces aliasing that causes a 6.97%–22.5% drop in F1 score. STFT computation time remains under 10 ms across all rates, negligible relative to the 1-second recording duration. *EfficientMic* achieves 70.28%–70.96% lower storage and 40.39%-51.14% lower energy consumption than the 44 kHz baseline, with only a modest 0.7%–1.16% drop in F1 score. While 16 kHz sampling shows similar performance, it consumes 19.92%–44.38% more energy and 27.6%–30.64% more storage than *EfficientMic*. Intermediate rates (e.g., 4 kHz and 8 kHz) show linearly scaling overheads but no meaningful F1 gain over 2 kHz, so we report only 2 kHz and 44 kHz in Figure 4 to clearly illustrate the trade-offs. Additionally, we evaluate the performance of *EfficientMic* on the LSTM architecture. LSTM achieves a 25% higher F1-score in ESC-50 and a 21% lower F1-score in the UrbanSound8K dataset. However, the memory consumption of an LSTM architecture is higher than a CNN model. Thus, we report the results of the CNN architecture. These results confirm that *EfficientMic* balances classification performance with resource efficiency. Differences in energy and storage impact across the two datasets further reveal how frequency content distribution shapes aliasing effects and system performance.

Different Algorithms for Aliasing Detection. Table 1 and Figure 5 compare *EfficientMic*'s aliasing detection module with different machine learning models, including Decision Tree (DT), Random Forest (RF), a dense neural network (DNN), a CNN with convolution and max pooling layers, and a two layer LSTM. Table 1 compares both the feature representation and aliasing detection model performance on an 80%-10%-10% train-validation-test framework at frequencies greater than 8 kHz. The machine learning algorithm chosen for *EfficientMic* is XGBoost as it outperforms all other models in terms of F1 score while maintaining low energy and latency. *EfficientMic* consistently achieves better F1-Score for aliasing detection across varying levels of aliasing in ESC-50, UrbanSound8k (Urban), and MAVD datasets, with a minimal time overhead of 0.09 ms. Additionally, *EfficientMic* consumes less energy than DNN or CNN models, though it uses slightly more energy than Decision Tree and Random Forest on both the ESC-50 and UrbanSound8K datasets. This demonstrates the system's ability to achieve the best accuracy without the highest energy consumption.

Time and Storage Overhead Analysis. We evaluate *EfficientMic*'s time and storage overhead compared to other machine learning models, including CNN, DNN, Decision Tree, and Random Forest. As shown in Figure 5, *EfficientMic* achieves the lowest inference latency (0.09–0.10 ms), outperforming Random Forest (0.28–0.33 ms), CNN (0.49–0.59 ms), and DNN (1.15–1.16 ms). Including STFT computation, the total latency remains under 20 ms, negligible relative to the 1-second audio window. Figure 2 illustrates how *EfficientMic* can minimize the duration of aliasing exposure. By reassessing aliasing every 20 ms, *EfficientMic* reduces the period of degraded recordings, improving downstream task performance. This allows task-specific trade-offs: calling *EfficientMic* less frequently saves power at the cost of more aliasing, while more frequent invocation minimizes aliasing at higher energy expense. These results show that *EfficientMic* achieves low latency and storage with minimal accuracy loss, ideal for real-time embedded and IoT systems.

5 Conclusion

This paper presents the *EfficientMic* framework, a novel approach for real-time aliasing detection in low-power embedded systems. By dynamically adjusting the microphone's sampling rate based on real-time analysis, *EfficientMic* balances both energy consumption and audio quality, addressing key challenges in resource-constrained settings such as long-lived and energy-harvesting sensor systems.

Evaluations on the UrbanSound8K and ESC-50 datasets show that *EfficientMic* significantly reduces storage and energy consumption, with only a minimal F1-Score drop-as low as 0.7%, outperforming traditional models. Since *EfficientMic* is designed for automated, on-device aliasing detection, we focus on quantitative metrics like accuracy, energy, and latency rather than STFT analysis. In the future, we plan to explore the effect of highly accurate aliasing detection on the quality of STFTs to determine the impact of aliasing suppression. Overall, *EfficientMic* offers an efficient, low-latency solution for real-time audio processing, making it well-suited for deployment in embedded and power-sensitive environments.

References

[1] L. W. Balling, L. D. Mosgaard, and D. Helmink. 2022. Signal processing and sound quality. *Hearing Review* 29, 2 (2022), 20–23.
[2] Chao Cai, Rong Zheng, and Jun Luo. 2022. Ubiquitous Acoustic Sensing on Commodity IoT Devices: A Survey. *IEEE Communications* (2022).
[3] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. ACM, 785–794.

Table 1: Average Macro-F1 score ± std across sample rates ≥8 kHz for each model and feature type in aliasing detection.

Dataset	STFT_xgb	STFT_rf	STFT_dt	STFT_lstm	MFCC_xgb	MFCC_rf	MFCC_dt	MFCC_lstm
ESC-50	$\mathbf{0.637 \pm 0.019}$	0.560 ± 0.019	0.538 ± 0.018	0.569 ± 0.047	0.275 ± 0.018	0.166 ± 0.008	0.307 ± 0.025	0.624 ± 0.010
Urban	$\mathbf{0.728 \pm 0.017}$	0.546 ± 0.003	0.707 ± 0.006	0.620 ± 0.014	0.679 ± 0.024	0.565 ± 0.012	0.661 ± 0.008	0.723 ± 0.009
MAVD	$\mathbf{0.787 \pm 0.024}$	0.366 ± 0.044	0.551 ± 0.025	0.334 ± 0.004	0.633 ± 0.036	0.524 ± 0.025	0.628 ± 0.019	0.705 ± 0.025

WASPAA25/Figures/ablation.pdf

Figure 5: *EfficientMic* outperforms other aliasing detection models on both ESC50 and UrbanSound8K datasets with 7.69-40.73% higher accuracy while consuming comparable energy. Average across all sample rates with K-Fold cross validation shown.

[4] Fabian Esqueda, Stefan Bilbao, and Vesa Valimaki. 2016. Aliasing reduction in clipped signals. *IEEE Transactions on Signal Processing* 64, 20 (2016), 5255–5267.

[5] Md Tamzeed Islam, Bashima Islam, and Shahriar Nirjon. 2017. SoundSifter: Mitigating Overhearing of Continuous Listening Devices. In *Proceedings of the 15th ACM International Conference on Mobile Systems, Applications, and Services*.

[6] Jinkee Kim, Yunyoung Jang, and Jong-Pal Kim. 2023. A Sound Activity Monitor with 96.3 μs Wake-up Time and 2.5 μW Power Consumption. *JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE* (2023).

[7] Vincent Lostanlen, Antoine Bernabeu, Jean-Luc Béchennec, Mikaël Briday, Sébastien Faucou, and Mathieu Lagrange. 2021. Energy Efficiency is Not Enough:Towards a Batteryless Internet of Sounds. In *Proceedings of the 16th International Audio Mostly Conference*. ACM.

[8] Karol J. Piczak. 2015. ESC: Dataset for Environmental Sound Classification. In *Proceedings of the 23rd Annual ACM Conference on Multimedia* (2015-10-13). ACM.

[9] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. 2014. UrbanSound8K.

[10] Leontiy K. Samoylov, Darya Yu. Denisenko, and Nikolay N. Prokopenko. 2018. The Function Approximation of the Signal Delay Time in the Anti-Alias Filter of the A/D Interface of the Instrumentation and Control System. In *2018 IEEE International Conference on Electrical Engineering and Photonics (EExPolytech)*.

[11] Leontiy K. Samoylov, Darya Yu. Denisenko, and Nikolay N. Prokopenko. 2019. Analog/Digital Anti-Aliasing Filters. In *2019 International Siberian Conference on Control and Communications (SIBCON)*.

[12] S. Ward-Foxton. 2025. PIMIC Adds Tiny AI to Microphones, Eyes Big AI Chips. *EE Times* (January 2025). Accessed: 2025-01-21.

[13] Pablo Zinemanas, Pablo Cancela, and Martín Rocamora. 2019. MAVD: A Dataset for Sound Event Detection in Urban Environments. In *Proceedings of the DCASE 2019 Workshop*. New York, USA, 25–26.