DOI: 10.61782/fa.2025.0518

FORUM ACUSTICUM EURONOISE 2025

LEVERAGING PRE-TRAINED FOUNDATIONAL AUDIO MODELS FOR ACOUSTIC EVENT DETECTION IN DRIVING SCENARIOS

Mohammad Moghimi [1*] **Patrick Healy** [1] **Carlos Castorena** [2]
Francesc J. Ferri [2] **Maximo Cobos**[2]
[1] Department of Computer Science, University of Limerick, Ireland
[2] Department of Computer Science, University of Valencia, Spain

ABSTRACT

Smart vehicles are experiencing increasing adoption, driven by a growing demand for their applications. To facilitate widespread deployment, it is important to improve the security and trust of these vehicles. This paper focuses on improving smart car safety by developing an audio-based system for detecting important sound events, including emergency vehicle sirens, tire skidding, car crashes, horn types, drifting, and others. Our methodology involves a multifaceted approach to classifying a diverse range of driving-related audio events. We begin with a Convolutional Recurrent Neural Network (CRNN) as a baseline. Subsequently, we investigate the performance of pre-trained foundational models (e.g., BEATs, Audio Spectrogram Transformers) followed by a Recurrent Neural Network layer, aiming to leverage the pre-trained representations for improved event classification. Additionally, we explore the potential benefits of combining these approaches, considering as well the effect of introducing augmented data. We propose novel hybrid models that integrate features extracted by the convolutional layers of the CRNN with those directly obtained from the pre-trained models. Our experimental results demonstrate significant performance gains when combining these distinct approaches into a unified architecture.

Keywords: *Sound Event Detection, Deep learning Models, Smart Vehicles, Driver safety, Convolutional Recurrent Neural Network (CRNN), Audio Spectrogram Transformer (AST), BEATs, Data Augmentation.*

1. INTRODUCTION

Human senses, especially sight and hearing, play a major role in our perception, communication, and safety. Although sight is undoubtedly the most important sense, hearing helps humans absorb and analyze surrounding information and may even be the only reliable source in certain situations. Due to the importance of auditory information, integrating auditory sensing into smart devices and autonomous systems, which are increasingly prevalent, has become essential [1]. These systems, including smart cars, are now equipped with various sensors such as cameras, long and short-range radars, Light Detection and Ranging (LiDAR), ultrasonic transducers, and GPS receivers. However, these sensors have limitations [1]. For instance, cameras have blind spots and are sensitive to lighting and scene structure, making visual information unreliable. According to the National Highway Traffic Safety Administration, approximately 840,000 blind spot incidents occur annually in the United States, resulting in 300 fatalities [2]. Similarly, lasers are ineffective in extreme weather conditions.

To handle these challenges, researchers are increasingly focusing on the ability of devices to automatically detect and classify sound events through technological means known as sound event detection (SED) [3]. SED is the automatic process that detects and classifies events in audio streams and estimates the onset and offset of these events [4]. SED has many applications, including audio/video surveillance [5,6], healthcare monitoring, environmental monitoring [7], industrial machine fault mon-

[*] ***Corresponding author***: *mohammad.moghimi@ul.ie.*

itoring [8], human-computer interaction, and smart cars. By accurately detecting and timestamping sound events, SED systems can enhance the capability, security, and trust in smart devices and autonomous systems.

However, SED is still an evolving research topic due to several challenges, such as having enough data for training, ensuring performance in noisy environments, real-time processing, and modeling overlapping sounds [9]. Additionally, the development of robust SED systems is complicated by the need for sufficiently labeled datasets with accurate annotations for training purposes, making them more applicable in real-world scenarios [10]. Traditional SED methods relied on manual feature extraction (e.g., MFCC [11], Mel spectrograms) and classical machine learning techniques such as SVM [11], KNN, and HMM/GMM [12]. While these approaches provided initial insights, they often required extensive experimentation and struggled in complex, noisy environments. These limitations spurred the transition to deep learning, which has revolutionized sound event detection. Recent works have shown that through deep learning models, specifically Convolutional Neural Networks (CNNs) [13,14] and Recurrent Neural Networks (RNNs) [15], state-of-the-art performance is achieved on numerous tasks involving pattern recognition and learning data representations. These models reduce the dependency on manual feature extraction. This gave deep learning models the potential to handle the complexities of multiple concurrent and overlapped sound events; hence deep learning has become the state-of-the-art approach for SED with better accuracy and robustness.

More recently, pre-trained foundational audio models, such as BEATs [16] and the Audio Spectrogram Transformer (AST) [17], have emerged as powerful tools for capturing universal audio representations from large-scale datasets. These models offer enhanced feature extraction capabilities and have shown promise in various audio analysis tasks, making them particularly well-suited for addressing the challenges of SED in dynamic environments like driving scenarios.

There are, however, a few remaining research gaps. Despite many advances, current models still need to better consider temporal dependencies, contextual complexities, and rich feature extraction. Moreover, more generalized models that perform well under diverse acoustic environments [15] are needed, and the issue of computational efficiency—especially for deployment in resource-constrained setups—remains a challenge. The main objective of this paper is to address these gaps by investigating novel deep learning architectures and training strategies aimed at developing more robust, general, and efficient SED systems for driving scenarios. In particular, this paper promotes the sound event detection field by developing and evaluating novel deep learning models that combine the strengths of traditional CNN and RNN architectures with the enhanced representations obtained from pre-trained embedding models. The critical contributions from this work include presenting a novel hybrid model that integrates convolutional and recurrent layers with pre-trained audio representations to improve both performance and generalization. Comprehensive experimental results on data collected in driving environments, demonstrate the effectiveness and superior performance of the proposed methods.

Section 2 provides an overview of the data used in our study. Section 3 outlines a baseline system and different proposed SED models. Section 4 describes the experimental setup, including the preprocessing step, training details, and evaluation metrics. Section 5 provides the experimental results and discusses their implications, and Section 6 concludes the paper with suggestions for future research.

2. DATASET

The dataset used in this study was collected by [18], specifically designed to support the training and evaluation of sound event detection systems in driving environments. This dataset has sounds that originated from both internal sources within the vehicle, such as vehicle components, devices, and human interactions, as well as external sources like traffic, road conditions, and environmental factors (see Fig. 1).

2.1 Dataset overview and Sources

The dataset comprises 19,000 audio files, each with a duration of 10 seconds, resulting in a total duration of approximately 53 hours. To ensure a balanced representation of events, the dataset was split into training, validation, and test subsets: 15,000 audio samples for training, 2,000 for validation, and 2,000 for testing [18]. The sound events were sourced from various publicly available datasets with unrestricted usage rights, including: Musical Genre Classification of Audio Signals [19], UrbanSound [20], AudioSet [21], Old Phone Ringtones as MIDI [22], ASR-CabNois (Cabin Noise Dataset) [23], Freesound [24], A Singing Voice Dataset [25], DCASE

Figure 1. Internal (e.g. ring tone) and external (e.g. horn, siren) sounds while driving.

2020 Task 4 Dataset [26], Donate-A-Cry Corpus Features Dataset [27].

2.2 Synthetic Data Generation and Labeling

To simulate realistic driving conditions, the dataset was created using **Scaper** [28] with augmentation techniques such as time stretching and pitch shifting applied to improve diversity and model generalization. Table 1 provides a concise overview of our dataset, which comprises 41 classes. In our classification framework, we explicitly consider a problem with 41 distinct classes. These events are organized into nine primary distractor categories- horns, Sirens, Speech, Physiological, Pets, Notifications, Vibrating, RingTone, and Cry- each of which represents sounds most likely to divert a driver's attention. All other environmental sounds, such as rain, thunder, or microphone noise, are grouped under "Other", thus completing the broader driving environment. Each event was assigned **hard labels**, providing precise onset and offset timestamps within the 10-second clips. The event annotations followed a standardized format: The audio file name, the event type, the event onset and offset times. This structured labeling approach ensured the dataset was compatible with multiple machine learning frameworks and facilitated accurate model training. The dataset is openly accessible via `https://www.kaggle.com/datasets/ccastorena/sound-event-detection-for-driver-safety` and can be utilized by researchers and practitioners for benchmarking and further development in sound event detection [18].

Table 1. Dataset Summary

Class Groups	Classes	Appearances
Horns	4	11,693
Sirens	3	8,858
Speech	1	3,020
Physiological	3	8,639
Pets	2	5,843
Notifications	1	2,862
Vibrating	1	2,988
RingTone	1	2,835
Cry	1	2,950
Other	24	16,597
Total	**41**	**66,285**

3. SED MODELS

Sound Event Detection (SED) involves identifying and localizing sound events by capturing both their spectral details and temporal dynamics. We begin with a popular CRNN baseline from the DCASE framework, which uses convolutional layers to extract features from Log-Mel spectrograms and recurrent layers to model how these features evolve over time. To further enhance our model's performance, we also combine additional feature extraction from pretrained models like the Audio Spectrogram Transformer (AST) and BEATs, creating a richer representation of the audio signals for more robust detection.

3.1 CRNN model as a baseline

We employ the DCASE baseline architecture [29] based on Convolutional Recurrent Neural Networks (CRNNs), which are widely used in sound event detection. This architecture is used to detect patterns in complex audio signals (via convolutional layers) and interpret how those patterns evolve over time (via recurrent layers). We use the standardized CRNN architecture defined by the DCASE framework, ensuring compatibility with other studies. As illustrated in Figure 2, the CRNN starts by converting audio into a Log-Mel spectrogram. This spectrogram, sized [1, 626, 128] (1 is the channel dimension, 626 is the time dimension, and 128 is the number of Mel frequency bands), is processed through seven convolutional blocks.

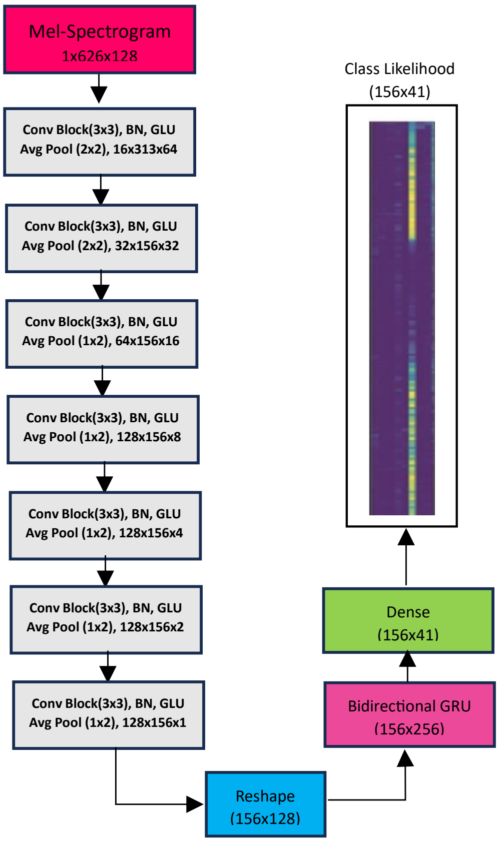

Figure 2. Baseline CRNN Architecture.

Each block uses a 2D convolution with a kernel size of 3×3, batch normalization, GLU activations, and average pooling. The number of filters in each layer is: 16, 32, 64, 128, 128, 128, 128, and the pooling operations has the specific dimensions of 2×2, 2×2, 1×2, 1×2, 1×2, 1×2, 1×2, respectively. This results in the convolutional output shape of [128, 156, 1]. To integrate with the bidirectional recurrent layers, the filter dimensions are transposed, and the extra dimension is removed. The bidirectional GRU layer produces an output of [156, 256], with 128 hidden units in both directions. Finally, a dense layer with softmax activation generates the final output with 156 frames and 41 distinct events.

3.2 Proposed models

Our proposed framework explores three distinct feature-extraction pathways for each audio clip, as illustrated in Figure 3. The first pathway, termed the *Baseline CNN Path*, uses Log-Mel spectrograms of size 1×626×128 as input to a convolutional neural network. This CNN processes the spectrograms to generate initial feature tensors with dimensions of [128, 156, 1].

The second pathway, referred to as the *AST Path*, leverages the frame-level version of the Audio Spectrogram Transformer (AST), specifically the ATST-Frame variant with a Base384 configuration. This model processes 10-second audio segments to produce a feature tensor of size 768×496 (representing the embedding dimension and time steps). An average pooling stage is applied to reduce the time dimension from 496 to match the CRNN's temporal resolution of 156, resulting in a pooled output of size [768, 156]. Features from this path are stored in HDF5 format for efficient loading.

The third pathway, known as the *BEATs Path*, utilizes BEATS_iter3_plus_AS2M.pt, a model pre-trained on AudioSet. Operating at a 16 kHz sampling rate with a 25 ms window (and a 10 ms hop length), this approach also outputs embeddings of size 768×496, which are similarly reduced via average pooling to produce a tensor of dimensions [768, 156]. Again, the features are stored in HDF5 format.

To harness the complementary strengths of these pathways, our framework investigates several fusion strategies. The simplest strategy is the CNN+RNN baseline, which relies solely on features extracted by the CNN. Alternatively, pretrained features from the BEATs and AST paths can be fed directly into recurrent layers (BEATs/AST Path + RNN). Another approach involves concatenating the CNN-extracted features with those from the AST or BEATs paths (CRNN + AST/BEATs), before inputting the combined representation into recurrent layers. Finally, a full integration strategy (CRNN + BEATs + AST) is proposed, combining all feature streams into a unified architecture. This multi-path fusion is designed to exploit the diverse and rich representations provided by each pathway, ultimately enhancing the robustness and accuracy of sound event detection in dynamic driving scenarios.

4. EXPERIMENTAL SETUP

4.1 Audio Pre-processing and Augmentation

All audio clips are first resampled to a 16 kHz sampling rate (mono) using Librosa. To ensure consistent input lengths, clips longer than 10 seconds are truncated, and those shorter than 10 seconds are zero-padded to reach

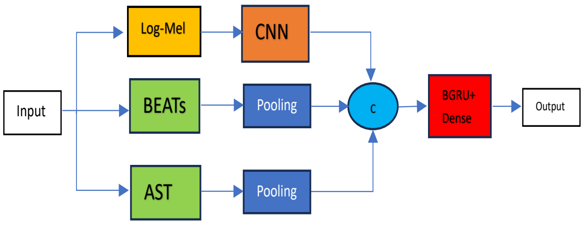

Figure 3. Proposed models.

the 10-second duration. For labeling, a frame-wise hard label format is used, where each frame corresponds to a discrete timestamp. Each frame has a value of 1 when an event is present and 0 otherwise. The output must predict 156 frames, corresponding to the chosen number of events. The model uses log-mel spectrograms as input features for the CNN model, using 128 mel-filters with a window length and FFT size of 2048, a Hamming windowing function, and a hope size of 256. The frequency range for the Mel spectrogram is from 0 Hz to 8000 Hz. The resulting Mel spectrogram has dimensions of 626 frames, 128 Mel bands, and a single channel. During training, we apply a combination of data augmentation techniques to enhance the robustness and generalization of our models. Specifically, we use the mixup technique and specaugment. Mixup is a widely adopted augmentation method in both machine listening and computer vision [30]. It creates synthetic training samples by linearly interpolating pairs of input spectrograms and their corresponding labels, thereby increasing the diversity of the training data and smoothing the decision boundaries. In addition, specaugment [31] is applied to the Log-Mel spectrograms, which involves randomly masking blocks of time and frequency bins. This perturbation forces the model to learn more invariant features and reduces sensitivity to noise, ultimately contributing to improved performance in dynamic and challenging driving scenarios.

4.2 Training Schedule

All experiments (CRNN-only, CRNN+AST, CRNN+BEATs, etc.) use the same overall training procedure: training continues for up to 200 epochs with early stopping triggered if no improvement is observed for 100 consecutive epochs, a dropout rate of 0.5 is applied, and the batch size is set to 48 for both training and validation. Additionally, the Adam optimizer is employed with a learning rate of 0.001 and a warm-up phase during the first 50 epochs.

4.3 Metrics

To comprehensively evaluate sound event detection systems, we employ a range of complementary metrics. While Accuracy is a common measure, relying on it exclusively can be misleading, especially for frame-by-frame detection and rare events. Thus, we complement it with more detailed metrics, such as the F1 score and the Polyphonic Sound Detection Score (PSDS).

The F1 score is widely used because it effectively balances Precision and Recall, thereby measuring the system's accuracy in detecting events without over- or under-detection. Meanwhile, the PSDS is computed using the `psds_eval` toolbox [32], which is a recognized standard within the DCASE community. PSDS values are derived from 50 operating points (linearly distributed from 0.01 to 0.99) and refer to the normalized area under the PSD-ROC (Receiver Operating Characteristic) curve up to a specified maximum effective false positive rate (eFPR).

In our experiments, we use PSDS1, where $DT_C = GT_C = 0.7$, in accordance with common practices in general SED tasks.

5. RESULTS AND DISCUSSION

Models	Augmentation	Accuracy	PSDS	F1
CRNN+ReLU	None	92	22.1	55.6
CRNN+ ReLU	Mixup+specaugment	94	23.5	56.3
CRNN+GLU(baseline)	Mixup+specaugment	95	28.1	58.1
AST+RNN	Non	96	30.1	59.9
BEATs+RNN	Non	98	31.5	60.6
CRNN+AST	Mixup+specaugment	97	37.4	61.2
CRNN+BEATs	Mixup+specaugment	97	39.7	61.6
BEATs+AST+RNN	Non	97	43.6	63.1
CRNN+BEATs+AST	Mixup+specaugment	97	44.3	63.4

Table 2. Comparison of models, augmentations, and their performance metrics.

Table 2 summarizes the performance of various sound event detection models. We begin with a CRNN+ReLU model without augmentation that achieved an accuracy of 92%, a PSDS of 22.1, and an F1 score of 55.6. When augmented with Mixup and specaugment, the CRNN+ReLU configuration improved slightly to 94% accuracy, 23.5 PSDS, and 56.3 F1. Replacing ReLU with GLU in the

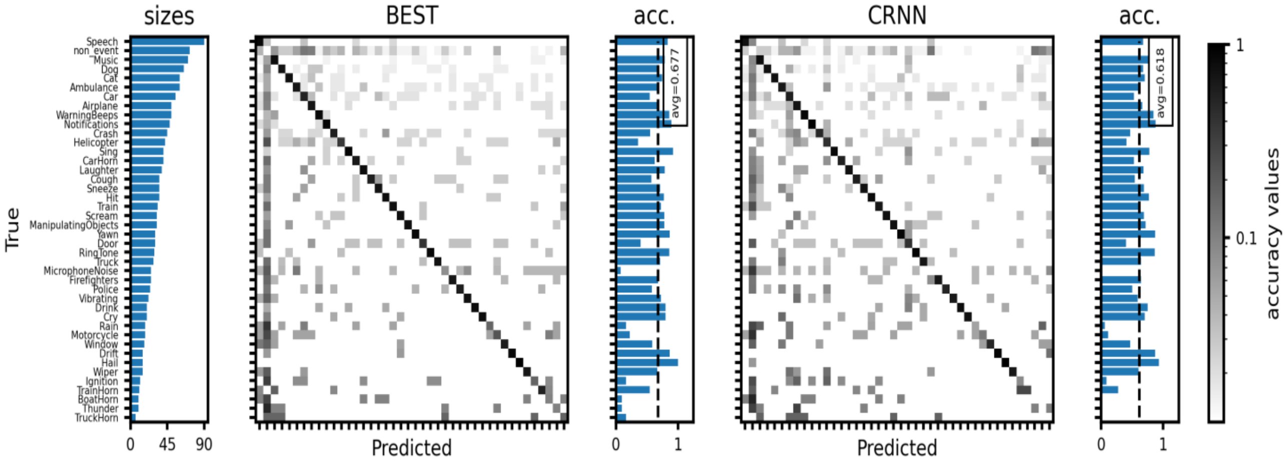

Figure 4. Confusion matrices comparing the best proposed model (left) and the CRNN+GLU baseline(right)

CRNN, while using the same augmentation, further enhanced performance to 95% accuracy, 28.1 PSDS, and 58.1 F1. This enhanced model not only serves as the baseline for the DCASE 2021-2023 editions [29], but it also becomes our reference point for comparing the performance of our proposed models, and forming the backbone of our hybrid approaches.

Models that integrate pre-trained audio features also showed notable improvements. The AST+RNN model, which does not use data augmentation, reached 96% accuracy, 30.1 PSDS, and 59.9 F1, while the Beats+RNN model outperformed it with 98% accuracy, 31.5 PSDS, and 60.6 F1.

The benefits of fusion strategies become even more evident in the hybrid models. The CRNN+AST and CRNN+Beats models, CNN employing Mixup and specaugment, achieved 97% accuracy with PSDS scores of 37.4 and 39.7, and F1 scores of 61.2 and 61.6, respectively. The Beats+AST+RNN configuration, without augmentation, reached 97% accuracy, 43.6 PSDS, and 63.1 F1. Finally, the fully integrated model CRNN+Beats+AST with Mixup and specaugment maintained 97% accuracy while achieving the highest PSDS (44.3) and F1 (63.4) scores.

To illustrate the behavior of the best model and the CRNN+GLU baseline on different classes, frame-based confusion matrices and associated accuracies are shown in Figure 4. The confusion matrices demonstrate that the BEST model has a stronger diagonal, reflecting a higher number of correct predictions, whereas the baseline shows more off-diagonal misclassifications. In Figure 4, we use a support-weighted per-class accuracy metric that differs from the frame-based accuracy presented in Table 2. The Table 2 metric is computed online using `torchmetrics`, evaluating the model's ability to correctly detect the presence or absence of sound events within fixed-length segments and capturing both active and inactive portions. By contrast, the metric in Figure 4 is derived post-hoc from a confusion matrix (often excluding the non-event class) by normalizing each class's counts and computing a weighted average based on class frequencies, effectively treating each class as a separate binary classification (one-vs-rest). These differences in aggregation, class inclusion, and weighting account for the observed discrepancies between the two accuracy values. Notably, for Figure 4, the BEST model achieves an average accuracy of 67.7% compared to the CRNN baseline's 61.8%, highlighting the benefits of integrating pre-trained audio features in driving scenarios.

These results indicate that leveraging pre-trained audio models substantially enhances the detection of complex, overlapping sound events in dynamic driving environments, with data augmentation further improving generalization. The superior performance of the hybrid models over the CRNN approaches underscores their potential for robust real-world sound event detection in challenging driving scenarios.

6. CONCLUSION

In this study, the integration of pre-trained models, specifically AST and BEATs, into a CRNN-based framework represents a significant breakthrough in sound event detection for driving scenarios. By leveraging the advanced feature extraction capabilities of models like AST and BEATs alongside the proven temporal modeling of CRNNs, our hybrid approach achieves superior performance—demonstrated by higher accuracy, PSDS, and F1 scores—compared to traditional methods. Data augmentation techniques such as mixup and specaugment further enhance model robustness in noisy, real-world environments, making the approach well-suited for the complex auditory landscapes encountered in smart vehicle applications.

Looking ahead, these findings open up promising avenues for future research, including real-time deployment optimizations, exploration of additional fusion strategies, and training on more diverse datasets. Overall, our work marks a significant step toward developing safer, more reliable autonomous systems by providing a robust framework for detecting critical sound events in dynamic driving conditions.

7. ACKNOWLEDGMENTS

This work has been supported by Science Foundation Ireland Grant No. 18/CRT/6223. Also, this work has been supported by Grant TED2021-131003B-C21 funded by MCIN/AEI/10.13039/501100011033 and by the “EU Union NextGenerationEU/PRTR”, as well as by Grant PID2022-137048OB-C41 funded by MICIU/AEI/10.13039/501100011033 and “ERDF A way of making Europe”.

8. REFERENCES

[1] E. G. Vidal, E. F. Zarricueta, and F. A. Cheein, “Human-inspired sound environment recognition system for assistive vehicles,” *Journal of Neural Engineering*, vol. 12, no. 1, p. 016012, 2015.

[2] W. H. Organization, *Global status report on road safety 2018*. World Health Organization, 2019.

[3] Z. Islam and M. Abdel-Aty, “Real-time emergency vehicle event detection using audio data,” 2022. arXiv preprint, arXiv:2202.01367.

[4] Z. Shuyang, T. Heittola, and T. Virtanen, “Active learning for sound event detection,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2895–2905, 2020.

[5] K. Manikanta, K. P. Soman, and M. S. Manikandan, “Deep learning based effective baby crying recognition method under indoor background sound environments,” in *Proceedings of the 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, pp. 1–6, IEEE, 2019.

[6] K. Deepak, S. Chandrakala, and C. K. Mohan, “Residual spatiotemporal autoencoder for unsupervised video anomaly detection,” *SIViP*, vol. 15, pp. 215–222, 2021.

[7] K. Deepak, G. Srivathsan, S. Roshan, and S. Chandrakala, “Deep multi-view representation learning for video anomaly detection using spatiotemporal autoencoders,” *Circuits, Systems and Signal Processing*, vol. 40, no. 3, pp. 1333–1349, 2021.

[8] M. Saimurugan and R. Ramprasad, “A dual sensor signal fusion approach for detection of faults in rotating machines,” *Journal of Vibration Control*, vol. 24, no. 12, pp. 2621–2630, 2018.

[9] A. Mesaros, T. Heittola, T. Virtanen, and M. D. Plumbley, “Sound event detection: A tutorial,” *IEEE Signal Processing Magazine*, vol. 38, no. 5, pp. 67–83, 2021.

[10] Q. Kong, Y. Xu, W. Wang, and M. D. Plumbley, “Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2450–2460, 2020.

[11] S. E. Küçükbay and M. Sert, “Audio-based event detection in office live environments using optimized mfcc-svm approach,” in *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, pp. 475–480, IEEE, 2015.

[12] B. Elizalde, M. Ravanelli, K. Ni, D. Borth, and G. Friedland, “Audio-concept features and hidden markov models for multimedia event detection,” in *SLAM@ INTERSPEECH*, pp. 3–8, 2014.

[13] H. C. Chu, Y. L. Zhang, and H. C. Chiang, “A cnn sound classification mechanism using data augmentation,” *Sensors*, vol. 23, no. 15, p. 6972, 2023.

[14] J. K. Das, A. Ghosh, A. K. Pal, S. Dutta, and A. Chakrabarty, "Urban sound classification using convolutional neural network and long short term memory based on multiple features," in *Proceedings of the 2020 Fourth International Conference On Intelligent Computing in Data Sciences (ICDS)*, pp. 1–9, IEEE, 2020.

[15] S. V. Shanmukha and K. Deepak, "Sound event detection in constrained real-world environments using fine tuned cnns," in *Proceedings of the 2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT)*, pp. 1–6, IEEE, 2024.

[16] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, and F. Wei, "Beats: Audio pre-training with acoustic tokenizers," *arXiv preprint arXiv:2212.09058*, 2022.

[17] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," *arXiv preprint arXiv:2104.01778*, 2021.

[18] C. Castorena, M. Cobos, J. Lopez-Ballester, and F. J. Ferri, "A safety-oriented framework for sound event detection in driving scenarios," *Applied Acoustics*, vol. 215, p. 109719, 2024.

[19] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[20] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proceedings of the 22nd ACM International Conference on Multimedia*, pp. 1041–1044, ACM, 2014.

[21] J. F. Gemmeke, D. P. W. Ellis, *et al.*, "Audio set: An ontology and human-labeled dataset for audio events," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 776–780, 2017.

[22] V. Torosyan, "Old phone ringtones as midi dataset," 2019. Online dataset.

[23] A. R. Team, "Asr-cabnois: A dataset of vehicle cabin noises for audio analysis," *Journal of Acoustic Research*, vol. 15, pp. 105–115, 2023.

[24] Freesound, "Freesound database," 2023. Online dataset.

[25] J. Wilkins, I. Mcloughlin, and D. Howard, "A singing voice dataset for mir research," *Journal of the Acoustical Society of America*, vol. 144, no. 3, pp. 1452–1463, 2018.

[26] R. Serizel, A. P. Shah, and J. Salamon, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *Workshop on Detection and Classification of Acoustic Scenes and Events (DCASE)*, 2020.

[27] A. Valani *et al.*, "Donate-a-cry corpus: A dataset of infant cries for sound classification," *Journal of Infant Research*, vol. 20, pp. 75–89, 2023.

[28] J. Salamon and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 344–348, IEEE, 2017.

[29] R. Serizel, N. Turpault, H. Eghbal-Zadeh, and A. Parag Shah, "Large-scale weakly labeled semi-supervised sound event detection in domestic environments," in *Workshop on Detection and Classification of Acoustic Scenes and Events*, (Woking, United Kingdom), Nov 2018. Submitted to DCASE2018 Workshop.

[30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[31] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," in *Interspeech 2019*, interspeech_2019, ISCA, sep 2019.

[32] Ç. Bilen, G. Ferroni, F. Tuveri, J. Azcarreta, and S. Krstulović, "A framework for the robust evaluation of sound event detection," in *ICASSP 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 61–65, IEEE, 2020.